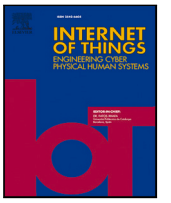
Internet of Things 26 (2024) 101209

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot



Research article

CICIoV2024: Advancing realistic IDS approaches against DoS and spoofing attack in IoV CAN bus

Euclides Carlos Pinto Neto [a], Hamideh Taslimasa [a], Sajjad Dadkhah [a,*], Shahrear Iqbal [b], Pulei Xiong [b], Taufiq Rahman [b], Ali A. Ghorbani [a]

[a] Canadian Institute for Cybersecurity - University of New Brunswick (UNB), Fredericton, New Brunswick, Canada
[b] National Research Council Canada, Ottawa, Ontario, Canada

ARTICLE INFO

Keywords:
Internet of Vehicles (ioV)
Internet of Things (ioT)
Intrusion Detection System (IDS)
Security
Dataset

ABSTRACT

Considering the complexity of network traffic in IoV operations, methods that can identify complex patterns become useful. Machine learning fosters several techniques to enhance the detection, prevention, and mitigation of cyberattacks. However, important features are not addressed in the current state-of-the-art security datasets for IoV. For example, in the case of intra-vehicle communications, it is critical to consider the interaction among multiple Electronic Control Units (ECUs). Also, mimicking a realistic IoV environment is not simple since establishing a test environment requires considerable financial investment. Hence, there is a need for a testbed composed of several real ECUs in an IoV environment comprising network traffic. Thereupon, the main goal of this research is to propose a realistic benchmark dataset to foster the development of new cybersecurity solutions for IoV operations. To accomplish this, five attacks were executed against the fully intact inner structure of a 2019 Ford car, complete with all ECUs. However, the vehicle was immobile and incapable of causing any potential harm or injuries. Hence, all attacks were carried out on the vehicle without endangering the car's driver or passengers. These attacks are classified as spoofing and Denial-of-Service (Dos) and were carried out through the Controller Area Network (CAN) protocol. This effort establishes a baseline complementary to existing contributions and supports researchers in proposing new IoV solutions to strengthen overall security using different techniques (e.g., Machine Learning — ML). The CICIoV2024 dataset has been published on CIC's dataset page.

1. Introduction

In the past few years, the Internet of Things (IoT) has become popular in serving society in different ways [1,2]. This paradigm brings various interconnected devices with sensing and actuation capabilities to optimize existing services in different areas [3,4]. There are several benefits of adopting IoT devices considering their simplicity to deploy and operate [5,6]. These devices allow different systems to be interconnected and share valuable information [7]. For example, in smart cities, solutions can be built across multiple sectors to increase urban efficiency and sustainability, such as transportation, healthcare, urban security, water supply, and energy [8,9]. IoT devices play a fundamental role in automation and control, such as factories adopt IoT devices to monitor production performance and to perform tasks with specific constraints [10]. This also enables the development of solutions for efficiency and energetic savings as IoT enables enhanced data collection [11]. Furthermore, IoT improves user satisfaction

* Corresponding author.
E-mail addresses: e.neto@unb.ca (E.C.P. Neto), h.taslimasa@unb.ca (H. Taslimasa), sdadkhah@unb.ca (S. Dadkhah), shahrear.iqbal@nrc-cnrc.gc.ca (S. Iqbal), pulei.xiong@nrc-cnrc.gc.ca (P. Xiong), taufiq.rahman@nrc-cnrc.gc.ca (T. Rahman), ghorbani@unb.ca (A.A. Ghorbani).

https://doi.org/10.1016/j.iot.2024.101209
Received 15 February 2024; Received in revised form 12 April 2024; Accepted 29 April 2024
Available online 7 May 2024
2542-6605/© 2024 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

and engagement and brings scalability and flexibility to businesses. New business opportunities involving IoT devices have been discovered in recent years. This paradigm is transforming the way businesses operate with remarkable examples in education [12], energy [13], and healthcare [14]. One of the main targets for IoT applications is vehicular technology, referred to as the Internet of Vehicles (IoV).

IoV is an extension of IoT that connects vehicles in an integrated smart automotive environment [15,16]. In this environment, vehicles interact with other services to improve the operation of transportation systems. Compared to traditional IoT applications such as agriculture and industrial IoT, IoV operates under specific automotive operational and technological constraints, including safety-critical aspects. Compared to traditional automotive operations, IoV presents several advantages in the short- and long-term [17]. For example, IoV enables real-time tracking of the vehicle's condition and identifies the need for preventive measures [18]. Predictive maintenance is widely supported by IoV devices since the deployment of multiple sensors can continuously monitor vehicles with high precision to anticipate mechanical problems that may occur [19]. Other supporting solutions are also made possible through the use of IoV devices [20]. Another area of services relies on the interaction between vehicles and external entities. For example, autonomous driving is extensively supported by IoV solutions and relies on the combination of control and perception [21]. Roadside assistance is another target for IoV deployments since environmental information can be collected in real-time and traffic [22]. Automotive businesses can also benefit from using IoV, including optimized fleet management and smart parking optimization [23]. Finally, IoV can potentially reduce emissions and make cars more environmentally friendly [24].

However, although IoT and IoV bring several advantages to existing systems, there is a critical concern regarding the cybersecurity aspects of these devices [25]. The number of cyber attacks targeting these systems has dramatically increased in the past few years [26], having the lack of computation resources and vulnerability standardization as key factors [27]. While these systems are easily integrated into existing topologies, complex attacks (e.g., Advanced Persistent Threats — APTs) have been targeting the fragility of their internal mechanism to get access to other resources in the network [28]. In terms of confidentiality, many devices in use in the market do not implement appropriate encryption techniques and may compromise privacy. Similarly, several devices present weak authentication mechanisms and, in many cases, default credentials. These vulnerabilities may allow malicious actors to capture and modify the data sent through the network, compromising the integrity of such services. Finally, these devices can be targeted by flood attacks that attempt to disrupt operations [29,30]. These threats can be critical depending on the device target, and disruptions can compromise the system's availability [31]. Although there have been new solutions developed lately for IoT and IoV security, there is still a critical demand for advanced solutions to prepare these systems for the complexity of new cyber threats [32].

Similarly to the general threats of IoT, many attacks are engineered and considered the IoV operational constraints [16]. In case of flooding attacks, vehicles may become unable to share and receive information from other entities [33]. Interception and modification of network traffic can result in misleading information being shared, which may lead to unsafe states [34]. Malicious actors may also focus on getting unauthenticated access to the system as well as exploiting physical vulnerabilities a vehicle may have [35]. The use of malicious software and injected network traffic can also compromise the vehicles' operation, while safety critical issues may arise from remote unauthorized access to the vehicles' controls. All these threats may lead to critical risks in the IoV operation. Also, privacy can be compromised in case some of these attacks are successfully launched [36]. The overall operational disruption can lead to various problems, such as damage, theft, and financial loss. To mitigate these issues, there is a current need for new solutions that can leverage the use of advanced techniques to improve the cybersecurity of IoV operations [16].

Considering the complexity and amount of network traffic in IoT and IoV operations, methods that can identify complex patterns become especially useful in these environments. Thus, Machine Learning (ML) fosters several techniques and solutions to enhance the detection, prevention, and mitigation of cyberattacks [37]. ML can detect abnormal traffic in the network, anomalous IoV traffic and requests, and aspects of IoV attacks [38,39]. These capabilities stem from the analysis of patterns in the network traffic presented in a multidimensional space. Potential threats can also be captured by ML, targeting to ensure that IoV can be used with a continuous security monitoring approach. Moreover, the specifications of the IoV network traffic can change depending on the application. For example, an autonomous driving IoV solution may present different network traffic patterns compared to an IoV weather service [40]. In all these cases, ML can be useful and help to advance the existing state-of-the-art by strengthening the protection IoV operations have. To ensure efficient solutions are developed, ML requires resources related to IoV operations. In this sense, one of the most important factors for an efficient solution is the dataset used to train the models.

Moreover, there are some datasets available in the literature for the development of security solutions for IoT and IoV. Conversely, there are important features not addressed in the current state-of-the-art contributions. For example, in the case of intra-vehicle communications, it is critical to consider the interaction among multiple ECUs. Also, mimicking a realistic IoV environment is not simple since establishing a test environment requires considerable financial investment. Hence, there is a need for a testbed composed of several real ECUs in an IoV environment comprising network traffic. IoV features can lead to advanced analytics methods to improve the security of automotive systems. Another critical aspect that needs to be considered in the production of a realistic dataset refers to the protocol used. In intra-vehicle communications, the CAN protocol [41] is the most used technology nowadays. To mimic an automotive IoV infrastructure, the topology needs to rely on CAN-BUS communication while multiple services are operable. Finally, the experiments of collection of the network traffic need to represent realistic scenarios, both in terms of attacks and in normal operations (i.e., benign traffic).

Thereupon, the main goal of this research is to propose a realistic benchmark dataset to foster the development of new cybersecurity solutions for IoV operations. To accomplish this, five attacks were executed against the fully intact inner structure of a 2019 Ford car, complete with all ECUs. However, the vehicle was rendered immobile and incapable of causing any potential harm or injuries. Hence, all attacks were carried out on the vehicle without endangering the car's driver or passengers. These attacks

are classified as spoofing and Denial-of-Service (DoS) and were carried out through the CAN-BUS protocol. This effort establishes a baseline complementary to existing contributions and supports researchers to propose new IoV solutions to strengthen the overall security using different techniques (e.g., Machine Learning — ML). Furthermore, the main contributions of this research are:

- **Development of a Comprehensive IoV Security Dataset:** This research addresses the critical gap in IoV cybersecurity by creating the CICIoV2024 dataset. This realistic benchmark dataset, derived from extensive experiments on a 2019 Ford vehicle's ECUs, offers a fine view of intra-vehicular communications, which is crucial for advancing cybersecurity solutions in IoV.
- **In-depth Analysis Using Machine Learning Techniques:** The paper provides a detailed evaluation of various ML algorithms, showcasing their efficacy in detecting, preventing, and mitigating cyberattacks in IoV systems. This analysis is essential in enhancing the understanding and application of ML in IoV cybersecurity.
- **New data characteristic to enhance IoV security solutions:** This effort introduces the analysis of ML performance using different data representations. In fact, the dataset is provided in binary, decimal, and hexadecimal formats. Furthermore, new attacks are executed in addition to those present in the existing state-of-the-art IoV security datasets.
- **Foundation for Future IoV Security Research:** The research sets a new baseline in IoV security, paving the way for future explorations. It opens avenues for further optimization of ML models, deeper feature analysis in IoV cybersecurity, and integration of the CICIoV2024 dataset with broader smart city systems. It lays the groundwork for developing diverse datasets focusing on different vehicle models.

The paper is organized as follows: Section 2 presents the main aspects of IoV applications in transportation and existing security datasets. Secondly, Section 3 introduces the CICIoV2024 dataset and depicts the phases involved in the data generation. Section 4 presents the method considered in this investigation to evaluate the data collected and Machine Learning (ML) algorithms. After that, Section 5 depicts the feature extraction process alongside a description of the data collected. Finally, Sections 6 and 7 present the ML evaluation in identifying different attacks and the conclusion of this research.

2. Literature review

In the past few years, the increase in IoV's popularity has fostered the development of new solutions focussing on the efficiency, security, and safety of transportation systems. This section reviews IoV applications in transportation alongside the work related to this research proposal.

2.1. IoV applications in transportation

IoV supports smart transportation systems in many ways, and twelve major areas are illustrated in Fig. 1. In terms of traffic surveillance, IoV brings many benefits to transportation systems. For example, IoV optimizes real-time traffic flow monitoring to increase performance and resiliency [42]. Also, congestion detection services become possible alongside advanced analysis on causes and consequences [43]. As different events may occur in urban environments, traffic pattern recognition services can also be developed, including incident detection capabilities [44]. Another important feature refers to the interaction of vehicles with external entities. For example, road condition monitoring and speed estimation are part of the IoV portfolio [45]. Regarding sustainability and compliance, IoV enables local environmental assessment and historical traffic analysis for policymakers [46]. Regarding a global view of the traffic conditions, solutions include vehicle counting and management [47]. In fact, traffic surveillance services can be extensively improved with the presence of IoV devices. Several types of devices can be deployed, including sensors and actuators. Furthermore, some IoV devices can be used to support existing systems while new devices are needed to enable the development of new transportation concepts [48].

Collision prevention is another critical area for IoV applications. Solutions include Vehicle-to-Vehicle (V2V) communication for proximity alerts [49] and optimized cruise control [50]. Some services are supportive of the general vehicle operations such as automated emergency braking [51], and blind zones and rear-end collision alerts [52]. Other solutions focus on the behavior of vehicles on the road, including driver fatigue and distraction monitoring [53], and vehicle speed monitoring and alerts [54]. All these aspects are critical to be considered alongside other nature-related events, for which services comprise inclement weather conditions warnings, vehicle stabilization and performance in adverse weather conditions [55], and potential road hazard warnings. All these aspects lead to the requirement of other IoV solutions, such as coordinated braking and steering control [56], Collision risk prediction, and obstacle detection. Finally, solutions are also available for emergencies and pedestrian detection.

Furthermore, IoV plays a vital role in developing and deploying self-driving cars. IoV-enabled solutions include autonomous navigation and route planning [57], and real-time pedestrian detection [58]. Computer vision is necessary for autonomous control, including traffic sign recognition, real-time traffic condition adaptation, and situational awareness. Maneuverability is another factor in autonomous systems since services such as self-parking [59], emergency [60], and overall optimization are part of the IoV capabilities. Safety is also a major concern for autonomous vehicles, and IoV supports safety improvement, remote monitoring, and fleet management [61]. Moreover, concerning sustainability, IoV presents a set of capabilities for green motivation. These capabilities comprise fuel-efficient methods [62] and car sharing [63]. Similarly, IoV supports the development of new solutions for driver support. Examples are augmented driving information and parking assistance [64]. Finally, augmented vision and collision avoidance systems [65] provide an improved experience for drivers.

Lightening is another important feature of driving. IoV brings solutions that include adaptive street and vehicle lighting [66]. In fact, lightning goes from pedestrian crossings to adaptations to adverse weather conditions. Similar solutions are applied to traffic

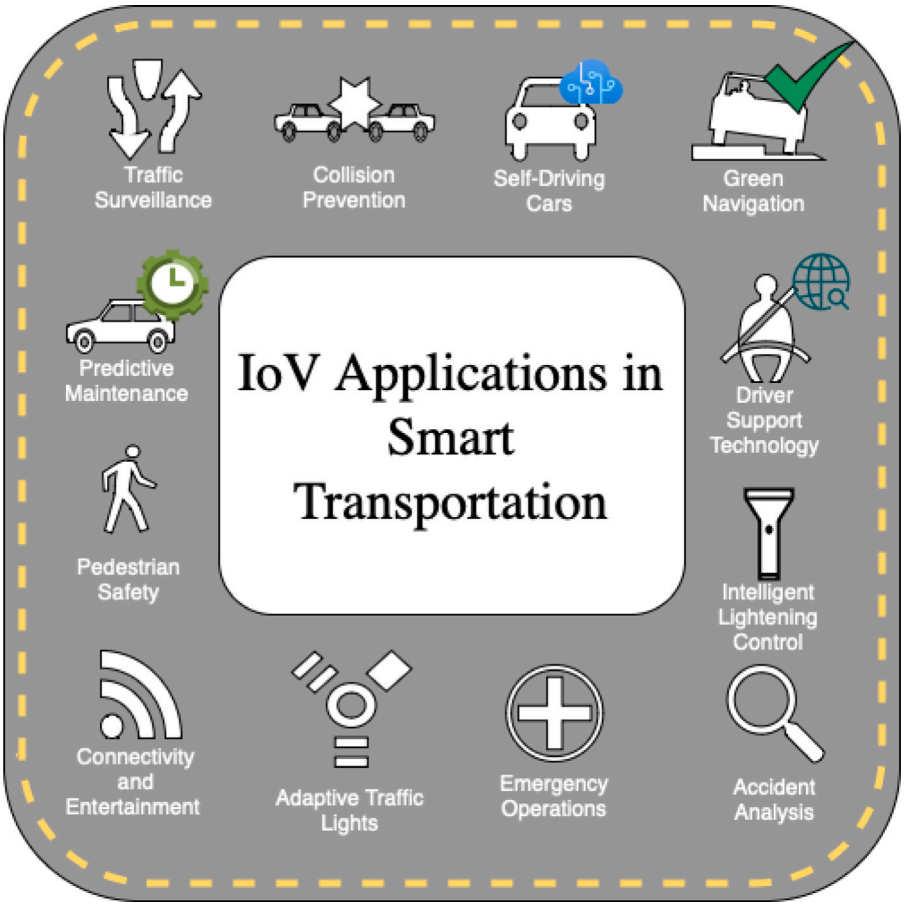


Fig. 1. Applications of IoV in smart transportation.

lights when adjustments can be made based on demand, and operations can be prioritized. Furthermore, another field for IoV applications is accident analysis. Several forensic factors must be considered in such assessments, and IoV can collect valuable data for these efforts. This data gathering relies on real-time crash reporting, accident reconstruction, and damage assessment based on automated diagnostics [67]. Hence, this is linked to the role of IoV in emergency management, where solutions optimize operations to increase efficiency.

Finally, services that surround vehicular operations are also supported by IoV. For example, there are several connectivity and entertainment solutions for new models. This connectivity also enables the deployment of predictive maintenance services, including many aspects of mechanical prognostics, fault prediction, oil and battery life prediction [68,69], and optimized maintenance schedule. All these aspects foster a safer environment and can contribute to pedestrian safety. There are solutions focused on pedestrians, including smart crosswalks and Vehicle-to-Pedestrian (V2P) communications [70].

Thereupon, IoV brings several benefits to transportation systems. Many aspects of vehicular operations can be improved and augmented with the use of these devices. This process results in a safer and more efficient environment, which represents a pillar of smart cities. There are still various issues in IoV technology that need to be addressed, and a major concern is cybersecurity.

2.2. Related works

The main purpose of this research is to advance the current state-of-the-art regarding IoV security datasets and support the development of new solutions in this field. Thereupon, the effort presented herein builds upon outstanding works published by different research groups. The authors in [71] introduce an IoV-specific dataset focusing on the use of CAN bus. In this investigation, a real Hyundai YF Sonata was used to conduct the attacks and collect the network traffic generated. The authors conduct multiple attacks against the vehicle, namely DoS attack, Fuzzy attack, RPM spoofing, and GEAR spoofing, and consider hexadecimal images as data representation. Additionally, the authors in [72] introduce the CAN Intrusion. In this case, DoS, fuzzy, and impersonation attacks were executed using a KIA Soul testbed and represented as hexadecimal data. Finally, the authors in [73] introduce a dataset containing IoV network traffic of many attacks, namely accelerator attack, correlated signal fabrication attack, fuzzing

Table 1
Comparison of the CICIoV2024 with existing efforts regarding attacks executed and data representation.

Work	Attacks performed	Data representation
GIDS	DoS attack Fuzzy attack RPM spoofing GEAR spoofing	Hexadecimal, Images
OTIDS	DoS attack Fuzzy attack Impersonation Attack Attack Free State	Hexadecimal
ROAD	Accelerator Attack Correlated Signal Fabrication Attack Fuzzing Attack Max Engine Coolant Temp Attack Max Speedometer Attack Reverse Light Off Attack Reverse Light OnAttack	Decimal
CICIoV2024	DoS Steering Wheel Spoofing RPM Spoofing Gas spoofing Speed Spoofing	Binary, Decimal, Hexadecimal

attack, max engine coolant temp fabrication/masquerade attack, max speedometer fabrication/masquerade attack, reverse light Off fabrication/masquerade attack, and reverse light On fabrication/masquerade attack. The authors adopt a decimal representation of the data collected. However, due to the limited number of IoV-specific security datasets available, many IoV contributions use general security and general IoT security datasets to validate their proposals. For example, several efforts used the CICIDS2017 [74] and the ISCXIDS2012 [75] datasets since it provides useful security traffic data referring to attacks such as brute force, DoS, DDoS, XSS, SQL injection, infiltration, botnet, and port scan [76–81]. Similarly, other security datasets are also adopted, e.g., NSL-KDD [82], UNSW-NB15 [83], ToNIoT [84], KDDCup99 [85], IoT BotNet [86], KDD99 [87], and AWID [88]. A comparison of the CICIoV2024 with existing efforts is presented in Table 1. Our research expands the existing state-of-the-art regarding the attacks executed, the testbed architecture adopted, and the evaluation of ML in multiple data representation methods.

3. The proposed CICIoV2024

This Section introduces the details of the process of producing the CICIoV2024. We present an overview of CAN bus and its packet structure. Then, we discuss the topology adopted and the characteristics of the real vehicle used. The process of sending messages to CAN bus is also presented, including how data was produced and collected for DoS, fuzzing, and spoofing attacks. In terms of benign traffic, the data was generated using the *candump* command, allowing random CAN bus data to be recorded without additional manipulation.

3.1. Controller Area Network (CAN) bus

The Controller Area Network (CAN) bus is one of the most popular intra-vehicle serial bus communication protocols [16]. Two serial buses compose the CAN bus — High-Speed (HS) and Low-Speed (LS) CAN (LS). High-Speed (HS) CAN presents a transmission rate of up to 1 Mbps, whereas Low-Speed (LS) CAN presents a rate ranging between 40 Kbps and 250 Kbps. ECUs are connected to both buses according to their respective priority [89].

Furthermore, each vehicle's component can access HS or LS randomly, considering that the CAN protocol uses a broadcast method to transmit data. Also, an arbitration method is adopted to ensure the importance and prioritization of critical and non-critical messages [90].

In this sense, HS CAN is responsible for transmitting data belonging to sensors and ECUs with tasks of higher priority. On the other hand, non-critical communications are established through the LS CAN bus. The CAN protocol and the CAN bus standard are widely adopted in most existing automated vehicles. In fact, this low-cost option simplifies the installation and deployment of solutions and services. Besides, it presents efficiency concerning communication overhead [91]. Finally, the low latency aspects and priority-based design allow the deployment of real-time applications (critical and non-critical).

However, there are serious security concerns in this scenario. CAN bus lacks authentication and encryption functions, which threatens the performance of real-time applications since it may be vulnerable to DoS and injection attacks. In the broadcast method adopted, the data sent lacks encryption, leading to the possibility of entities present in the communication (both legitimate and malicious) injecting data into the bus. Thus, CAN bus is a vulnerable asset that attackers can exploit. Efforts have been targeting CAN bus vulnerabilities and security issues in the intra-vehicle domain. These efforts aim at addressing security shortcomings of this protocol from multiple points of view [92,93].

Table 2
CAN Packet Structure.

Start of frame (SOF)	Arbitration (ID)	R T R	Control field	Data field	Cyclic redundancy check (CRC)	Acknowledgment (ACK)	End of frame (EOF)
1 bit	11 or 29 bits	1 bits	6 bits	64 bits	16 bits	2 bits	7 bits

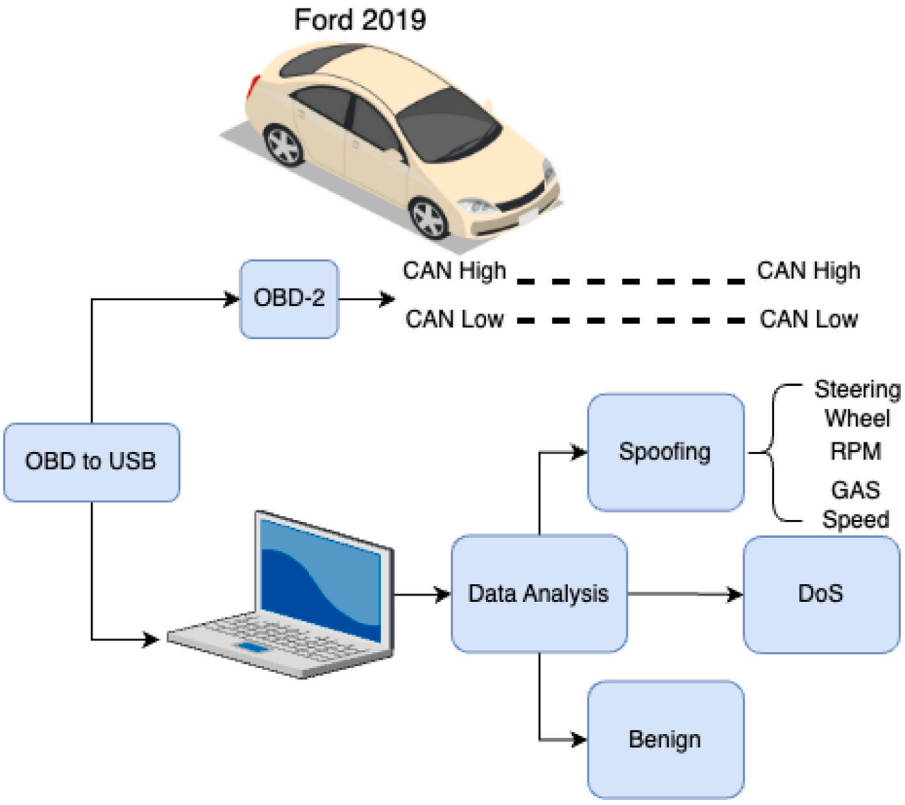


Fig. 2. Testbed adopted for the CICIoV2024 dataset.

3.2. CAN packet structure

The CAN protocol presents a straightforward packet structure illustrated in Table 2. The Start of Frame (SOF) and End of Frame (EOF) mark the beginning and end of a packet. Information regarding CAN controller is provided by the Cyclic Redundancy Check (CRC) and Acknowledgment (ACK) field. The arbitration field in the CAN bus determines which node can control the bus and transmit data and which message to receive according to their arbitration ID filter. It contains an 11-bit or 29-bit identifier and 1 bit dedicated to Remote Transmission (RTR). The ID field is also responsible for the priority mechanism of CAN, where smaller arbitration ID defines higher priority and vice versa. Another important field in the CAN packet is data bytes which are the actual messages being transmitted among nodes. Thereupon, we present different attack scenarios in CAN bus and how they are performed in a real vehicle. The data generated can be used to further implement defense solutions as well as enhance in-vehicle network protocols.

Furthermore, several nodes capable of transmitting data can be deployed into a vehicle. For this reason, fields in the CAN protocol stack enable the categorization and priority of each message. For example, the ID field refers to the message priority in eight bytes or less. If multiple nodes try to send data simultaneously, the message with a higher priority prevails, illustrating the operation of a priority-based arbitration. In this case, lower ID values represent higher priority and have preference in transmission. As the CAN bus comprises two buses (LS and HS), multiple devices can be connected to both of them and have access to the three most important parts of the CAN frame: Arbitration Identifier, Data Length Code, and Data field.

3.3. Access CAN bus in a real car

This research was conducted using a real car to ensure the data collected is realistic. Fig. 2 presents the topology used to produce the CICIoV2024 dataset focuses on the execution of attacks and collection and analysis of the network traffic captured. Besides,

Fig. 3. Inner structure of a 2019 Ford vehicle, complete with all ECUs.

Fig. 3 illustrates the fully intact inner structure of the 2019 Ford car used in this research, complete with all ECUs. The vehicle was rendered immobile and incapable of causing any potential harm or injuries. Hence, all attacks were carried out on the vehicle without endangering the car's passengers. Adopting this car model enables the evaluation of vehicles with similar architectures, even those manufactured by different companies. The Onboard Diagnostic port, also known as the OBD-II port, serves as the most direct interface to a vehicle's CAN bus. In the past, hacking a car through the OBD-II port has faced criticism for being an unrealistic attack method, as it assumes physical access to the port. However, modern connected cars possess a broader attack surface with various potential entry points. While remote attacks on vehicles receive more media attention, conducting research through direct access to the OBD-II port remains a valid approach to understanding how a vehicle can be manipulated after gaining access. Similarly, any manipulation possible via OBD-II can be performed remotely on vehicles with remote CAN connectivity. OBD-II is located next to the seats and can be accessed even without the use of special tools. Ultimately, the CICIoV2024 enables the development of solutions to support smart cities in the context of smart transportation safety and security. In terms of IoV attack detection, our proposal empowers solutions for similar vehicular architectures and enables the adoption of multiple data representations to uncover underlying malicious patterns.

The execution of attacks and collection of the network traffic was made possible through the use of dedicated hardware and software. In terms of hardware, we adopt a USB2CAN device and the ELM327 as a Bluetooth-based device. Also, the Macchina M2 was used, which is an open-source automotive interface that allows systems to communicate to the CAN bus via OBD-II. Macchina M2 is modular and allows the addition of Wi-Fi, GSM, LTE, and BLE modules on top of M2. M2 has 2 CAN channels and LIN.

In terms of software, *SocketCAN*, *can-utils*, *vcan* built into the Linux kernel are the tools adopted. They serve the purpose of sending and receiving, encoding, and decoding the CAN packets. Moreover, Wireshark [94] can analyze CAN packets. The analysis and transmission of CAN packets is made possible through the use of *CAN-utils*, a Linux-specific set of utilities that enables communication with the CAN network on the vehicle. Furthermore, the *canutils* consists of 5 main tools: *cansniffer*, responsible for sniffing packets; *cansend*, which is capable of writing packets; *candump*, used to dump all packets received; *canplayer*, used to replay CAN packets; and *cangen*, that enables the generation of random CAN packets.

Moreover, *SocketCAN* can be used as an interface to the CAN bus network. The virtual CAN driver (*vcan*) is the virtual CAN interface that allows CAN frame sharing without real CAN controller hardware. The common names given to virtual CAN network devices start with 'vcanX' (e.g., *vcan0*, *vcan1*, and *vcan2*). For a real vehicular network, *vcan* is replaced by *can*. For example, *vcan0*, *vcan1*, and *vcan2* would be named as *can0*, *can1*, ..., *can2*). Finally, setting the bitrate is also possible.

3.4. Sending messages to CAN bus

To send messages to CAN bus, one can use the *can-utils* library for Linux, comprising a *cansend* tool for sending payloads. However, it is possible to face specific CAN bus errors in communication. For example, a device may turn off its bus state if a large number of CAN-bus-related errors occur. In that case, communication is interrupted and frames stop to be sent and received. Conversely, an automatic recovery procedure can be used by adopting a non-zero value to the "restart-ms" parameter. Furthermore, the generation of dummy packets is also possible through the use of *can-utils*.

In terms of sniffing, we can use *cansniffer*, a tool provided by *can-utils* capable of capturing and sniffing packets. This tool enables the analysis of changes in CAN traffic and can be used to evaluate particular bytes of interest. Thus, *cansniffer* allows the colored observation of changes by comparing previous and current bytes. Based on these features, it is possible to know what operations and commands have been sent in the car.

Moreover, making sense of the data collected through CAN bus is complex. Besides, identifying the arbitration ID for frame injection is a difficult task. In this case, malicious actors may sniff the packets and change internal values for specific purposes. This approach highlights the vulnerability of CAN communication. Since it is possible to evaluate if malicious actions could affect the vehicle operation, attackers can use different evaluation mechanisms. Thereupon, the frames captured and saved by *candump* can be replayed using *canplayer*.

Car hacking carries inherent dangers that must never be overlooked. When dealing with a vehicle's CAN bus, it is crucial to remember that the target system is a heavy metal object capable of reaching high speeds swiftly. Unlike typical computer hacking, where errors may result in operating system corruption, a Blue Screen of Death, or data loss, car hacking could lead to severe injuries or even fatalities. Therefore, it is imperative to approach car hacking with caution and under controlled conditions. As the CAN bus contains critical control units of the vehicle, experimenting with CAN messages may unintentionally trigger responses from the engine, brakes, transmission, or other components. Even if the engine or transmission is not the intended focus, preparing for potential worst-case scenarios is vital. Safety measures should always be a top priority when hacking cars.

3.5. Exploitation

The experiments considered in this research include multiple attacks focussed on the CAN bus operation. Our main goal is to propose a realistic benchmark dataset to support designing new cybersecurity solutions for IoV operations. Despite being the most used protocol for intra-vehicle networks, CAN bus presents several security issues related to confidentiality and authentication. This research considers two classes of threats against CAN bus: DoS and spoofing attacks.

3.5.1. Denial-of-Service (DoS)

The possibility of manipulating the arbitration mechanism allows attacks to be executed. For Denial-of-Service (DoS), malicious CAN bus users can alter the arbitration values to launch attacks. For example, an attacker can launch a DoS campaign by flooding the system with misleading high-priority packets, preventing legitimate communication from being successfully established [95]. Also, manipulating the transmission rate adopted for a particular communication flow can prevent ECUs from accessing the bus, compromising the system's availability [93].

The lack of authentication methods for CAN bus operations empowers attackers to transmit random CAN frames to ECUs and store the outcome [96]. The intra-vehicle network becomes unstable since unexpected and unwanted network behaviors can be triggered by the manipulation of CAN frames. In fact, this threat enables a tailored analysis of the impact such CAN frames can have on different ECUs.

Performing a DoS attack on the Controller Area Network (CAN) bus involves intentionally flooding the bus with a large number of messages or interfering with the normal communication process to disrupt its functionality. In this case, we perform arbitration ID spoofing to overload the CAN Bus. The attacker continuously transmits messages with the same Arbitration ID as critical control messages. Since the CAN bus uses the Arbitration ID to prioritize messages, flooding the bus with higher priority messages can disrupt regular communication and delay or prevent essential commands from being delivered. DoS attack also can be done using fuzzy attack with the difference that priority will be set to lower priority nodes.

Table 3
Spoofing attack configuration.

CAN ID	D0	D1	D2	D3	D4	D5	D6	D7	Comments	Manipulated bytes
0 × 80	SWH	SWL							Steering Wheel SW — steering wheel position in 0.1 degrees (90 degrees = 900)	Data = {D0: 0 × 84, D1: 0 × 03} 90 degrees Data = {D0: 0 × C2, D1: 0 × 01} 45 degrees
0 × 201	RPMH	RPML							RPM — engine RPM multiplied by 4 (4000=1000 RPM)	Data = {D0: 0 × A0, D1: 0 × 0F} (1000 RPM) Data = {D0: 0 × 40, D1: 0 × 1F} (2000 RPM)
0 × 201					Speed				Speed — divided by 2 (50 =100 km/h)	Data = {D4: 0 × 64} (200 km/h) Data = {D4: 0 × 32} (100 km/h)
0 × 201							GASH	GASL	GAS — gas pedal position (range 0-0 × c800)	Data = {D6: 0 × 40, D7: 0 × 9C} Data = {D6: 0 × 00, D7: 0 × 7D}

3.5.2. Spoofing attack

Spoofing, also called impersonation attacks, occurs when an attacker gains access to the CAN Bus network and intercepts all the transmitted traffic. The features observed include CAN ID, payload range, and transmission rate [97]. This knowledge enables attackers to pretend to be legitimate nodes by spoofing the arbitration ID, leading to potential attacks aimed at disabling specific ECUs. For example, the attacker could intercept packets related to door openings and inject malicious data while the vehicle is in motion. Similarly, the attacker might manipulate Revolutions Per Minute (RPM) data during driving, posing significant risks to everyone onboard [16].

To execute spoofing attacks, precise details about the targeted ECUs are required. However, publicly available information about cars is limited due to privacy concerns unless such information has been previously investigated and documented in the literature through experimental hacking. In our study, we rely on information from [98,99] to compile relevant data concerning these attacks. Table 3 illustrates the crucial packet information we can utilize to manipulate parameters such as the steering wheel, RPM, speed, and gas control. Finally, spoofing and DoS attacks differ in several ways, including the attack process, the characteristics of the network traffic, and the different impact on the system's CIA (Confidentiality, Integrity, and Availability) triad.

4. Methodology

After conducting the attacks and collecting the traffic, the data stored needs to be organized. Fig. 4 illustrates the process of storing the data for different experiments (i.e., benign, DoS, and spoofing - steering wheel, RPM, gas, and speed), converting the data to binary and decimal values and storing them into CSV files, defining train and test sets, conducting the Machine Learning (ML) evaluation, and presenting the integrated results.

First, after collecting data from the CAN bus, the samples are stored in a TXT format, which requires further cleansing and preprocessing. In fact, the numerical data collected is stored in hexadecimal. After that, to facilitate future investigation, we provide both binary and decimal representations of the same data, in addition to the original hexadecimal format. This process consists of a row-based conversion of values, in which the number of features does not change for the decimal representation. However, the binary representation increases the number of features since each feature stores a bit to represent hexadecimal value.

Then, we split the dataset into two parts for each attack and dataset representation (i.e., decimal and binary). The first part contains 80% of the data collected and is used to train ML models. The second part refers to the train set and comprises 20% of the data captured. This division enables evaluating different ML models, and we evaluate the performance of widely used algorithms. Logistic Regression [100] is a lightweight approach that uses multiple variables to predict the outcome. Random Forest [101] is an ML method that generates random subsets of training configurations and makes decisions based on the evaluation of several perspectives [102]. Adaboost [103–105] is an ensemble model that aligns the answer of underlying models to make global decisions. These weak models can use different algorithms, and the hyperparameters enable Adaboost to change its internal execution in many ways. Deep Neural Networks [106] represent an ML model that mimics the human brain by adopting multiple neurons. These models have demonstrated outstanding success in several applications and can be used to solve complex cybersecurity problems. These models were chosen based on their popularity and wide adoption. Future directions of this research include the evaluation of other ML algorithms in the context of IoV security. Table 4 presents the parameters adopted by each model.

Finally, we present the results in an integrated way using multiple metrics. Assuming *TP* True Positives, *TN* True Negatives, *FP* False Positive, and *FN* False Negatives, the metrics used to evaluate are as follows:

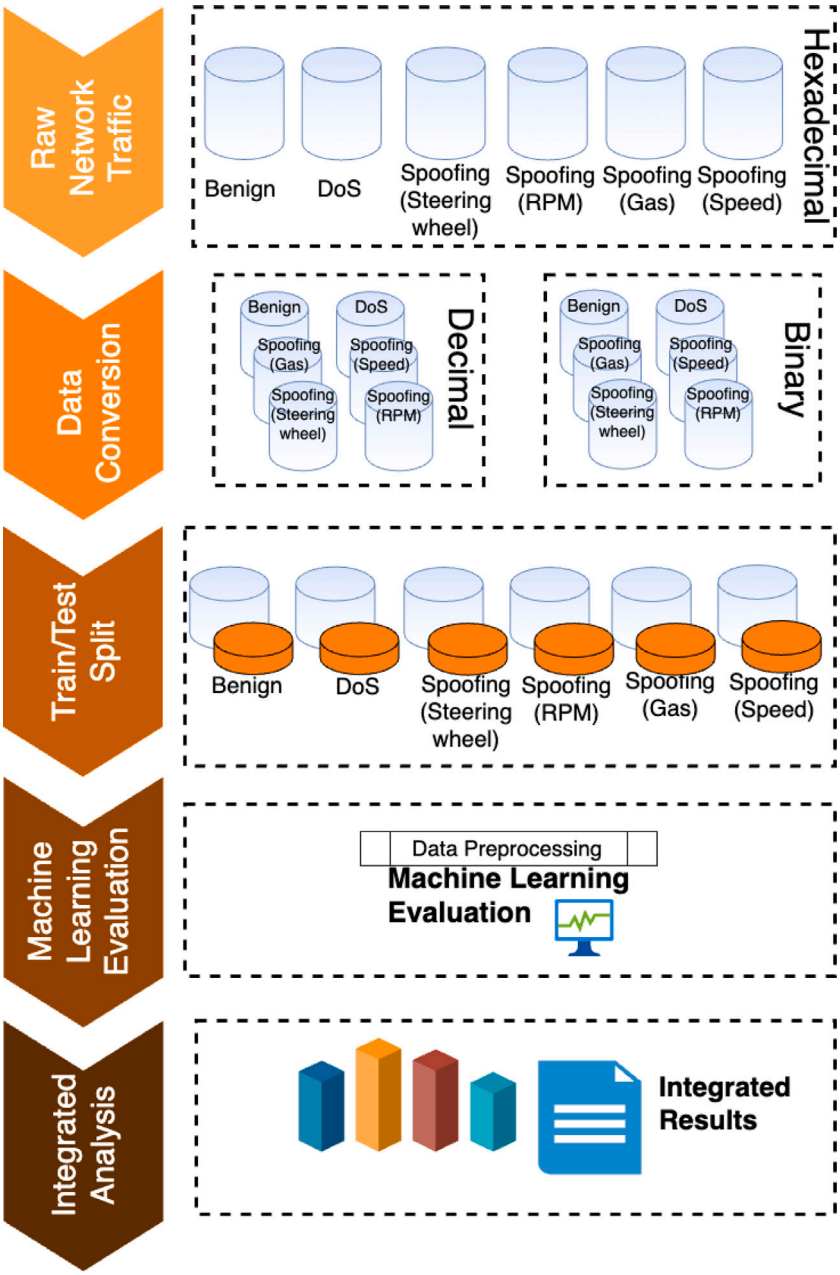


Fig. 4. Process of evaluating ML algorithms using raw network traffic.

- **Accuracy:** assess the classification models through the estimation of the portion of correct predictions as follows:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

- **Recall:** ratio of label correctly identified to the total number of that particular label in the dataset:

$$Rec = \frac{TP}{TP + FN} \tag{2}$$

- **Precision:** ratio of instances correctly classified to the total number of positive classifications:

$$Pre = \frac{TP}{TP + FP} \tag{3}$$

Table 4
Parameters used for each ML model.

Model	Parameters
Random Forest (RF)	n_estimators = 100, criterion = 'gini', min_samples_split = 2, min_samples_leaf = 1, min_weight_fraction_leaf = 0.0, max_features = 'sqrt', min_impurity_decrease = 0.0, bootstrap = True, oob_score = False, warm_start = False, ccp_alpha = 0.0
AdaBoost (AB)	estimator = DecisionTreeClassifier, n_estimators = 50, learning_rate = 1.0, algorithm = 'SAMME.R'
Logistic Regression (LR)	penalty = 'l2', dual = False, tol = 0.0001, C = 1.0, fit_intercept = True, intercept_scaling = 1, solver = 'lbfgs', max_iter = 100, multi_class = 'auto', warm_start = False
Deep Neural Network (DNN)	hidden_layer_sizes = (16,16,16,16), solver = 'adam', alpha = 0.0001, batch_size = 'auto', learning_rate = 'constant', learning_rate_init = 0.001, power_t = 0.5, max_iter = 200, shuffle = True, tol = 0.0001, warm_start = False, momentum = 0.9, nesterovs_momentum = True, early_stopping = False, validation_fraction = 0.1, beta_1 = 0.9, beta_2 = 0.999, epsilon = 1e−08, n_iter_no_change = 10, max_fun = 15000

Table 5
Features extracted for the CICIoV2024 dataset.

Feature name	Description
ID	Arbitration — indicates the priority of the message and the type of data it carries.
DATA_0	Byte 0 of the data transmitted.
DATA_1	Byte 1 of the data transmitted.
DATA_2	Byte 2 of the data transmitted.
DATA_3	Byte 3 of the data transmitted.
DATA_4	Byte 4 of the data transmitted.
DATA_5	Byte 5 of the data transmitted.
DATA_6	Byte 6 of the data transmitted.
DATA_7	Byte 7 of the data transmitted.
label	The identification of benign or malicious traffic.
Category	The identification of the category to which the traffic belongs.
Specific_class	The identification of the specific class of the traffic.

- **F1-Score:** geometric average of recall and precision:

$$F1 = 2 \times \frac{Pre \times Rec}{Pre + Rec} \tag{4}$$

5. Features and data description

Extracting features from intra-vehicular communication simplifies threat detection. Table 5 lists the features available on the decimal dataset. The first feature (*ID*) refers to arbitration. Then, the data features (i.e., *DATA_0*, ..., *DATA_7*) represent the bytes of the data transmitted. Finally, *label*, *category*, and *specific_class* are classifications for the CAN bus traffic. The binary dataset adopts the same approach. However, the data is divided in bits instead of bytes. Tables 6, 7, and 8 describe the CICIoV2024 data for binary and decimal representations. These tables list the mean, standard deviation (std), minimum (min), 25% percentile (25%), 50% percentile (50%), 75% percentile (75%), and maximum (max) values for each feature. Table 6 presents descriptive statistics for the decimal representation of the CICIoV2024 dataset. The data fields' maximum and minimum values are the same since they represent the storage capacity, which changes regarding the arbitration field. Finally, Tables 7 and 8 present descriptive statistics for the binary representation of the CICIoV2024 dataset. In this case, rather than having an aggregated field for each feature, we separate them to show their bit-wise values, which vary between 0 and 1.

Thereupon, Fig. 5 illustrates the class distribution for (a) labels, (b) Categories, and (c) specific classes (spoofing). In the experiments conducted, it was possible to collect attacks and benign traffic. Among the attacks, spoofing was conducted more extensively than DoS due to the possible scenarios. In fact, RPM spoofing was conducted and traffic was collected, alongside speed, steering wheel, and gas spoofing. Finally, Table 9 depicts the number of instances available on the dataset for each class.

Table 6
Data description for the decimal dataset.

	Mean	Std	Min	25%	50%	75%	Max
ID	537.207946	322.479994	65	357	516	578	1438
DATA_0	71.0865995	88.9771748	0	0	16	127	255
DATA_1	69.9892503	95.5837431	0	0	12	128	255
DATA_2	55.0112724	72.7658378	0	0	13	125	255
DATA_3	57.4536382	90.3207664	0	0	0	92	255
DATA_4	45.2851673	64.4583498	0	0	6	86	255
DATA_5	53.8826134	94.3361202	0	0	0	63	255
DATA_6	71.7491441	101.687183	0	0	0	138	255
DATA_7	60.2747691	99.9654672	0	0	0	80	255

Table 7
Data description for the binary dataset (Part I).

Feature	Mean	Std	Min	25%	50%	75%	Max	Feature	Mean	Std	Min	25%	50%	75%	Max
ID0	0.00	0.00	0	0	0	0	0	DATA_116	0.30	0.46	0	0	0	1	1
ID1	0.00	0.00	0	0	0	0	0	DATA_20	0.00	0.00	0	0	0	0	0
ID2	0.00	0.00	0	0	0	0	0	DATA_21	0.00	0.00	0	0	0	0	0
ID3	0.00	0.00	0	0	0	0	0	DATA_22	0.00	0.00	0	0	0	0	0
ID4	0.00	0.00	0	0	0	0	0	DATA_23	0.00	0.00	0	0	0	0	0
ID5	0.00	0.00	0	0	0	0	0	DATA_24	0.00	0.00	0	0	0	0	0
ID6	0.17	0.38	0	0	0	0	1	DATA_25	0.00	0.00	0	0	0	0	0
ID7	0.43	0.49	0	0	0	1	1	DATA_26	0.00	0.00	0	0	0	0	0
ID8	0.29	0.45	0	0	0	1	1	DATA_27	0.00	0.00	0	0	0	0	0
ID9	0.13	0.34	0	0	0	0	1	DATA_28	0.00	0.00	0	0	0	0	0
ID10	0.38	0.48	0	0	0	1	1	DATA_29	0.17	0.37	0	0	0	0	1
ID11	0.50	0.50	0	0	1	1	1	DATA_210	0.23	0.42	0	0	0	0	1
ID12	0.48	0.50	0	0	0	1	1	DATA_211	0.31	0.46	0	0	0	1	1
ID13	0.19	0.39	0	0	0	0	1	DATA_212	0.25	0.44	0	0	0	1	1
ID14	0.57	0.49	0	0	1	1	1	DATA_213	0.34	0.47	0	0	0	1	1
ID15	0.47	0.50	0	0	0	1	1	DATA_214	0.31	0.46	0	0	0	1	1
ID16	0.53	0.50	0	0	1	1	1	DATA_215	0.31	0.46	0	0	0	1	1
DATA_00	0.00	0.00	0	0	0	0	0	DATA_216	0.33	0.47	0	0	0	1	1
DATA_01	0.00	0.00	0	0	0	0	0	DATA_30	0.00	0.00	0	0	0	0	0
DATA_02	0.00	0.00	0	0	0	0	0	DATA_31	0.00	0.00	0	0	0	0	0
DATA_03	0.00	0.00	0	0	0	0	0	DATA_32	0.00	0.00	0	0	0	0	0
DATA_04	0.00	0.00	0	0	0	0	0	DATA_33	0.00	0.00	0	0	0	0	0
DATA_05	0.00	0.00	0	0	0	0	0	DATA_34	0.00	0.00	0	0	0	0	0
DATA_06	0.00	0.00	0	0	0	0	0	DATA_35	0.00	0.00	0	0	0	0	0
DATA_07	0.00	0.00	0	0	0	0	0	DATA_36	0.00	0.00	0	0	0	0	0
DATA_08	0.00	0.00	0	0	0	0	0	DATA_37	0.00	0.00	0	0	0	0	0
DATA_09	0.22	0.42	0	0	0	0	1	DATA_38	0.00	0.00	0	0	0	0	0
DATA_010	0.34	0.48	0	0	0	1	1	DATA_39	0.19	0.39	0	0	0	0	1
DATA_011	0.32	0.47	0	0	0	1	1	DATA_310	0.26	0.44	0	0	0	1	1
DATA_012	0.33	0.47	0	0	0	1	1	DATA_311	0.24	0.43	0	0	0	0	1
DATA_013	0.31	0.46	0	0	0	1	1	DATA_312	0.29	0.45	0	0	0	1	1
DATA_014	0.38	0.48	0	0	0	1	1	DATA_313	0.27	0.44	0	0	0	1	1
DATA_015	0.32	0.47	0	0	0	1	1	DATA_314	0.29	0.45	0	0	0	1	1
DATA_016	0.30	0.46	0	0	0	1	1	DATA_315	0.27	0.44	0	0	0	1	1
DATA_10	0.00	0.00	0	0	0	0	0	DATA_316	0.28	0.45	0	0	0	1	1
DATA_11	0.00	0.00	0	0	0	0	0	DATA_40	0.00	0.00	0	0	0	0	0
DATA_12	0.00	0.00	0	0	0	0	0	DATA_41	0.00	0.00	0	0	0	0	0
DATA_13	0.00	0.00	0	0	0	0	0	DATA_42	0.00	0.00	0	0	0	0	0
DATA_14	0.00	0.00	0	0	0	0	0	DATA_43	0.00	0.00	0	0	0	0	0
DATA_15	0.00	0.00	0	0	0	0	0	DATA_44	0.00	0.00	0	0	0	0	0
DATA_16	0.00	0.00	0	0	0	0	0	DATA_45	0.00	0.00	0	0	0	0	0
DATA_17	0.00	0.00	0	0	0	0	0	DATA_46	0.00	0.00	0	0	0	0	0
DATA_18	0.00	0.00	0	0	0	0	0	DATA_47	0.00	0.00	0	0	0	0	0
DATA_19	0.28	0.45	0	0	0	1	1	DATA_48	0.00	0.00	0	0	0	0	0
DATA_110	0.24	0.43	0	0	0	0	1	DATA_49	0.12	0.32	0	0	0	0	1
DATA_111	0.29	0.45	0	0	0	1	1	DATA_410	0.26	0.44	0	0	0	1	1
DATA_112	0.25	0.43	0	0	0	0	1	DATA_411	0.18	0.39	0	0	0	0	1
DATA_113	0.32	0.47	0	0	0	1	1	DATA_412	0.24	0.42	0	0	0	0	1
DATA_114	0.35	0.48	0	0	0	1	1	DATA_413	0.25	0.43	0	0	0	1	1
DATA_115	0.28	0.45	0	0	0	1	1	DATA_414	0.30	0.46	0	0	0	1	1

Table 8
Data description for the binary dataset (Part II).

Feature	Mean	Std	Min	25%	50%	75%	Max	Feature	Mean	Std	Min	25%	50%	75%	Max
DATA_415	0.40	0.49	0	0	0	1	1	DATA_67	0.00	0.00	0	0	0	0	0
DATA_416	0.32	0.47	0	0	0	1	1	DATA_68	0.00	0.00	0	0	0	0	0
DATA_50	0.00	0.00	0	0	0	0	0	DATA_69	0.27	0.44	0	0	0	1	1
DATA_51	0.00	0.00	0	0	0	0	0	DATA_610	0.30	0.46	0	0	0	1	1
DATA_52	0.00	0.00	0	0	0	0	0	DATA_611	0.33	0.47	0	0	0	1	1
DATA_53	0.00	0.00	0	0	0	0	0	DATA_612	0.24	0.43	0	0	0	0	1
DATA_54	0.00	0.00	0	0	0	0	0	DATA_613	0.29	0.45	0	0	0	1	1
DATA_55	0.00	0.00	0	0	0	0	0	DATA_614	0.24	0.43	0	0	0	0	1
DATA_56	0.00	0.00	0	0	0	0	0	DATA_615	0.36	0.48	0	0	0	1	1
DATA_57	0.00	0.00	0	0	0	0	0	DATA_616	0.30	0.46	0	0	0	1	1
DATA_58	0.00	0.00	0	0	0	0	0	DATA_70	0.00	0.00	0	0	0	0	0
DATA_59	0.17	0.38	0	0	0	0	1	DATA_71	0.00	0.00	0	0	0	0	0
DATA_510	0.25	0.43	0	0	0	0	1	DATA_72	0.00	0.00	0	0	0	0	0
DATA_511	0.23	0.42	0	0	0	0	1	DATA_73	0.00	0.00	0	0	0	0	0
DATA_512	0.23	0.42	0	0	0	0	1	DATA_74	0.00	0.00	0	0	0	0	0
DATA_513	0.31	0.46	0	0	0	1	1	DATA_75	0.00	0.00	0	0	0	0	0
DATA_514	0.31	0.46	0	0	0	1	1	DATA_76	0.00	0.00	0	0	0	0	0
DATA_515	0.34	0.47	0	0	0	1	1	DATA_77	0.00	0.00	0	0	0	0	0
DATA_516	0.40	0.49	0	0	0	1	1	DATA_78	0.00	0.00	0	0	0	0	0
DATA_60	0.00	0.00	0	0	0	0	0	DATA_79	0.22	0.41	0	0	0	0	1
DATA_61	0.00	0.00	0	0	0	0	0	DATA_710	0.26	0.44	0	0	0	1	1
DATA_62	0.00	0.00	0	0	0	0	0	DATA_711	0.25	0.43	0	0	0	1	1
DATA_63	0.00	0.00	0	0	0	0	0	DATA_712	0.25	0.44	0	0	0	1	1
DATA_64	0.00	0.00	0	0	0	0	0	DATA_713	0.22	0.42	0	0	0	0	1
DATA_65	0.00	0.00	0	0	0	0	0	DATA_714	0.22	0.41	0	0	0	0	1
DATA_66	0.00	0.00	0	0	0	0	0	DATA_715	0.24	0.43	0	0	0	0	1
								DATA_716	0.23	0.42	0	0	0	0	1

Table 9
Number of instances for each class present in the CICIoV2024 dataset.

Label	Category	Class	Count
Benign	–	–	1 223 737
Attack	DoS	–	74 663
	Spoofing	GAS	9991
		Steering Wheel	19 977
		Speed	24 951
		RPM	54 900

Table 10
Results obtained in the Machine Learning (ML) evaluation.

		LogisticRegression	AdaBoost	DeepNeuralNetwork	RandomForest
Binary	Accuracy	0.95	0.87	**0.95**	**0.95**
	Recall	**0.68**	0.17	**0.68**	**0.68**
	Precision	**0.74**	0.14	**0.74**	0.60
	F1-score	**0.63**	0.15	**0.63**	0.62
Decimal	Accuracy	0.89	0.92	**0.96**	**0.96**
	Recall	0.50	0.66	**0.76**	**0.76**
	Precision	0.48	0.48	**0.83**	0.76
	F1-score	0.49	0.51	**0.78**	0.76

6. Machine learning (ML) evaluation

The experiments conducted in this research assessed the performance of four ML algorithms: Logistic Regression (LR), Random Forest (RF), Adaboost (AB), and Deep Neural Network (DNN). Two classification problems were considered, using the (i) decimal and (ii) binary representations. In both cases, the models are expected to classify instances into benign, DoS, Steering Wheel (SW) spoofing, gas spoofing, RPM spoofing, and speed spoofing. Fig. 6 illustrates the results obtained for all scenarios (2 classes, 3 classes, and 6 classes) for both binary and decimal representations regarding accuracy, recall, precision, and F1-score.

Most models perform well for the classification task involving two classes, especially when adopting a binary encoding (a). These results change when the classification task separates the malicious activities into different classes (b), highlighting that only DNN and RF are able to perform well in both binary and decimal encodings. Finally, the overall evaluation metrics of all models are reduced when a more granular separation of malicious activities is adopted (c).

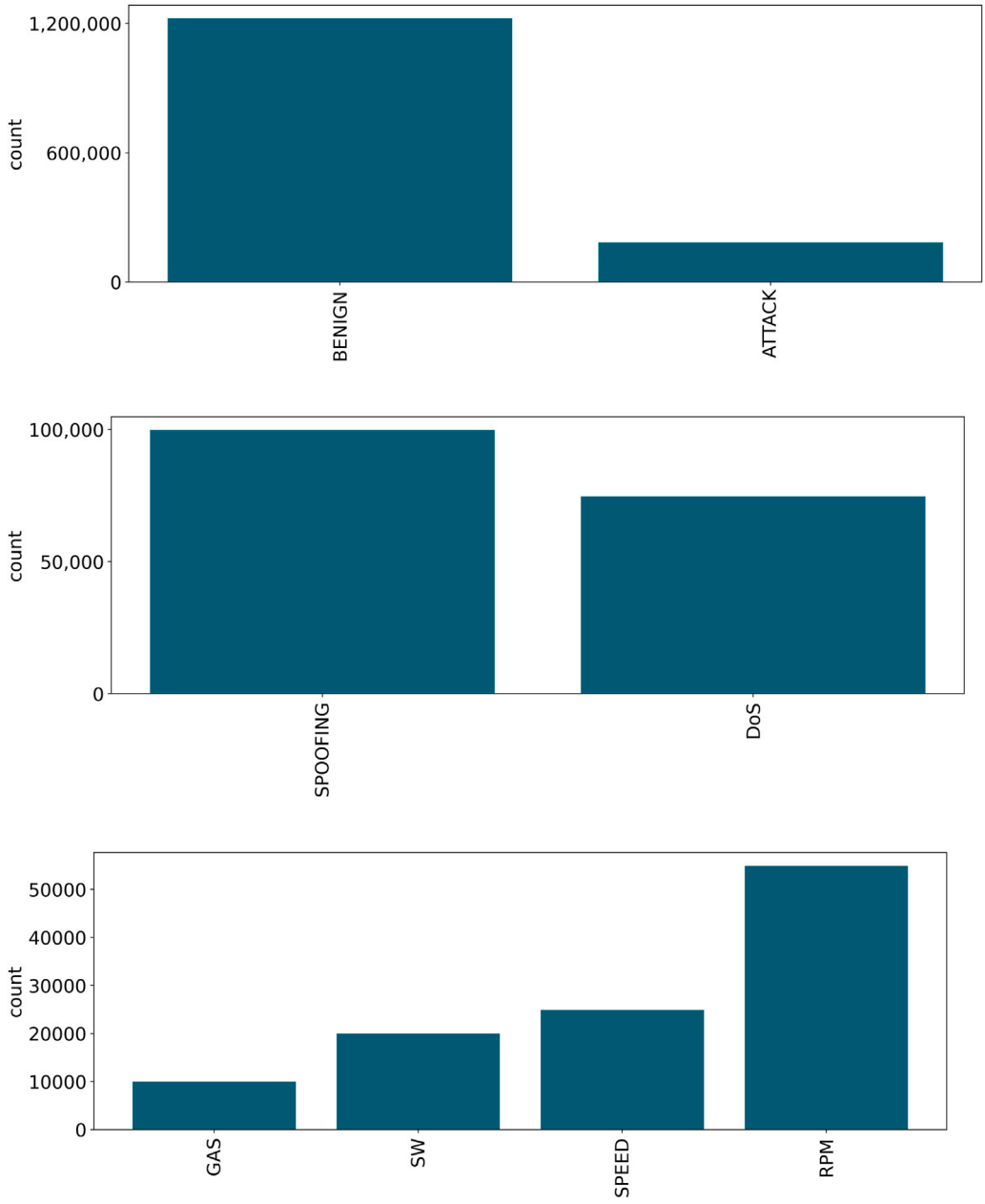
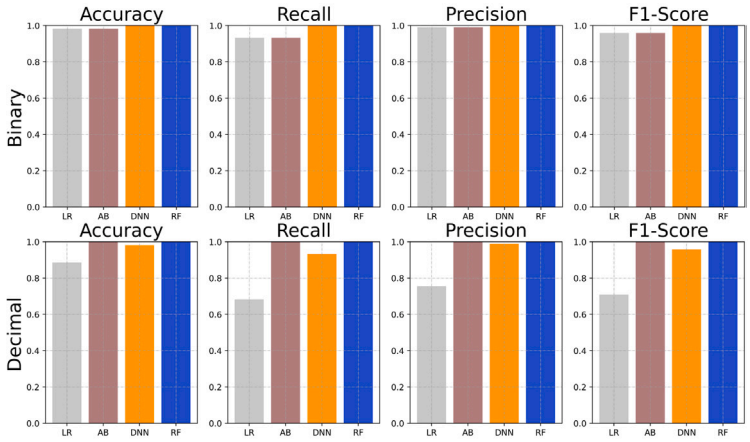
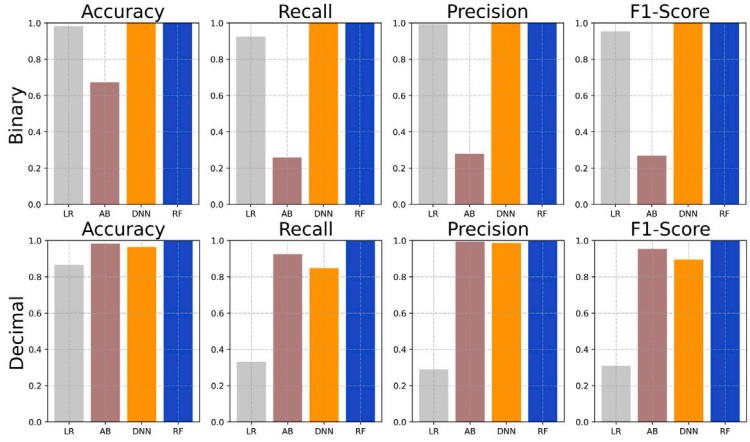


Fig. 5. Class Distribution for (a) labels, and (b) Categories, and (c) specific classes (spoofing)

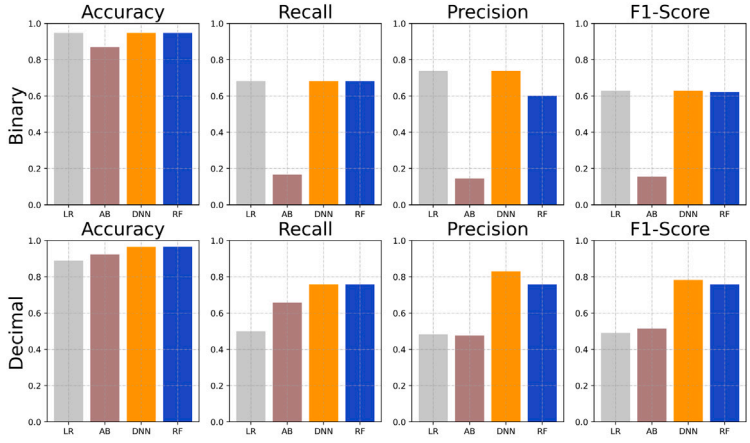
The results show that for binary classification (2 classes) and categorical classification (3 classes), some techniques achieved a perfect score (1.0). This shows that the attack traffic presents different patterns to the benign collection. Furthermore, for the multiclass classification problem (6 classes), DNN and RF outperformed the other methods. In all cases, the accuracy achieved was high. However, the performance evaluation for each class demonstrates that DNN and RF present higher F1 scores. Table 10 lists the numerical results of the multiclass classification problem. Although DNN presented an overall high performance, DNN and RF models presented similar results for binary and decimal representations. Both outperformed LR and AB, especially in terms of precision and F1-score. Table 11 presents the confusion matrix for all methods for binary and decimal datasets (6 classes). In this matrix, the lines represent the true labels and are illustrated by the first and last columns to simplify visualization. The columns represent the predicted labels and are illustrated at the top of each matrix. Overall, the models can identify the right labels in many cases. However, AB is not capable of classifying attacks in binary representation. Besides, there is an overlapping among spoofing



(a) Binary Classification (Benign and Attack)



(b) Categorical Classification (Benign, DoS, and spoofing)



(c) Multiclass classification (i.e., benign, DoS, gas spoofing, steering wheel spoofing, speed spoofing, and RPM spoofing)

Fig. 6. Results obtained in the Machine Learning (ML) evaluation.

attacks (SW, SPEED, RPM) in many cases due to the similarity in traffic. The main challenge faced during the training and testing these ML models was the performance discrepancy when the number of classes changed. A more granular classification decreases the overall performance, presenting a more severe impact on recall, precision, and F1-score. Due to the traffic characteristics and the imbalanced nature of attack datasets, the models perform well for some classes and present problems with others. For example, DoS

Table 11
Confusion matrix for the multiclass classification problem.

			Benign	DoS	SW	Speed	RPM	Gas	
Binary	Benign	AB	244 748	0	0	0	0	0	Benign
	DoS		14 933	0	0	0	0	0	DoS
	SW		3996	0	0	0	0	0	SW
	Speed		4991	0	0	0	0	0	Speed
	RPM		10 980	0	0	0	0	0	RPM
	Gas		1999	0	0	0	0	0	Gas
	Benign	LR	244 748	0	0	0	0	0	Benign
	DoS		1	14 932	0	0	0	0	DoS
	SW		0	0	3996	0	0	0	SW
	Speed		4989	0	0	2	0	0	Speed
	RPM		5	0	4991	4989	995	0	RPM
	Gas		0	0	0	0	0	1999	Gas
	Benign	DNN	244 748	0	0	0	0	0	Benign
	DoS		0	14 933	0	0	0	0	DoS
	SW		0	0	3996	0	0	0	SW
	Speed		4989	0	0	2	0	0	Speed
	RPM		0	0	4991	4989	1000	0	RPM
	Gas		0	0	0	0	0	1999	Gas
	Benign	RF	244 748	0	0	0	0	0	Benign
	DoS		0	14 933	0	0	0	0	DoS
	SW		0	0	3996	0	0	0	SW
	Speed		0	0	0	2	4989	0	Speed
	RPM		0	0	4991	4989	1000	0	RPM
	Gas		0	0	0	0	0	1999	Gas
			Benign	DoS	SW	Speed	RPM	Gas	
Decimal	Benign	AB	232 308	0	1728	0	10 712	0	Benign
	DoS		1	14 909	0	23	0	0	DoS
	SW		3996	0	0	0	0	0	SW
	Speed		0	0	0	2	0	4989	Speed
	RPM		0	0	0	0	10 975	5	RPM
	Gas		0	0	0	0	0	1999	Gas
	Benign	LR	244 395	0	0	191	162	0	Benign
	DoS		14 933	0	0	0	0	0	DoS
	SW		1	0	3995	0	0	0	SW
	Speed		4991	0	0	0	0	0	Speed
	RPM		10 980	0	0	0	0	0	RPM
	Gas		0	0	0	0	0	1999	Gas
	Benign	DNN	244 748	0	0	0	0	0	Benign
	DoS		0	14 933	0	0	0	0	DoS
	SW		0	0	3996	0	0	0	SW
	Speed		4989	0	0	2	0	0	Speed
	RPM		5	0	0	4989	5986	0	RPM
	Gas		0	0	0	0	0	1999	Gas
	Benign	RF	244 748	0	0	0	0	0	Benign
	DoS		0	14 933	0	0	0	0	DoS
	SW		0	0	3996	0	0	0	SW
	Speed		0	0	0	2	4989	0	Speed
	RPM		0	0	0	4989	5991	0	RPM
	Gas		0	0	0	0	0	1999	Gas

is a class that all models easily identify. However, speed spoofing is the most challenging attack to identify. Finally, data collected from realistic IoV operations are naturally imbalanced, i.e., benign traffic is expected to be more common than malicious traffic. For this reason, the CICIoV2024 mimics a real operation and enables ML models to train under these constraints. Future directions of this research include the development of strategies to effectively train ML models in an imbalanced setting.

7. Conclusion

This research focuses on the development of an intra-vehicular communication security dataset called CICIoV2024 to enhance IoV security. To accomplish this, six attacks were executed against the fully intact inner structure with all ECUs of a 2019 Ford car. This effort establishes a baseline complementary to existing contributions and supports researchers in proposing new IoV solutions to strengthen overall security using different techniques.

Throughout the paper, a comprehensive discussion was conducted to explain how the dataset was collected, processed, represented, and stored. An evaluation of different ML algorithms was also conducted, considering different classification tasks. We presented a detailed overview of how each attack was conducted, how features were extracted and used, and described the data collected. The dataset is available in hexadecimal, decimal, and binary representations and comprises the CAN bus traffic collected throughout the experiments.

There are various possible directions for future work. For example, the evaluation and optimization of additional ML models, the in-depth analysis of each feature in the classification, and the integration of the CICIoV2024 dataset with other components of the smart city paradigm. Finally, developing other datasets focusing on different vehicle systems is also in the scope of future works.

CRediT authorship contribution statement

Euclides Carlos Pinto Neto: Writing – original draft, Visualization, Software, Methodology, Data curation, Conceptualization. **Hamideh Taslimasa:** Writing – original draft, Visualization, Software, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Sajjad Dadkhah:** Writing – review & editing, Validation, Supervision, Resources, Project administration. **Shahrear Iqbal:** Writing – review & editing, Supervision, Project administration. **Pulei Xiong:** Writing – review & editing, Project administration, Investigation, Conceptualization. **Taufiq Rahman:** Data curation, Conceptualization. **Ali A. Ghorbani:** Validation, Supervision.

Declaration of competing interest

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

Data availability

The data will be shared in the page of the institution.

Acknowledgments

The authors graciously acknowledge the support from the Canadian Institute for Cybersecurity (CIC), the funding support from the National Research Council of Canada (NRC) through the AI for Logistics collaborative program, the NSERC Discovery Grant (no. RGPIN 231074), and Tier 1 Canada Research Chair to Dr. Ghorbani.

References

[1] S. Li, L.D. Xu, S. Zhao, The internet of things: a survey, Inf. Syst. Front. 17 (2015) 243–259.
[2] L. Tan, N. Wang, Future internet: The internet of things, in: 2010 3rd International Conference on Advanced Computer Theory and Engineering, ICACTE, 5, IEEE, 2010, pp. V5–376.
[3] M. Kocakulak, I. Butun, An overview of Wireless Sensor Networks towards internet of things, in: 2017 IEEE 7th Annual Computing and Communication Workshop and Conference, CCWC, Ieee, 2017, pp. 1–6.
[4] N. Khalil, M.R. Abid, D. Benhaddou, M. Gerndt, Wireless sensors networks for Internet of Things, in: 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing, ISSNIP, IEEE, 2014, pp. 1–6.
[5] S. Pundir, M. Wazid, D.P. Singh, A.K. Das, J.J. Rodrigues, Y. Park, Intrusion detection protocols in wireless sensor networks integrated to Internet of Things deployment: Survey and future challenges, IEEE Access 8 (2019) 3343–3363.
[6] D.G. Korzun, S.I. Balandin, A.V. Gurtov, Deployment of Smart Spaces in Internet of Things: Overview of the design challenges, in: Conference on Internet of Things and Smart Spaces, Springer, 2013, pp. 48–59.
[7] M.B. Mollah, M.A.K. Azad, A. Vasilakos, Secure data sharing and searching at the edge of cloud-assisted internet of things, IEEE Cloud Comput. 4 (1) (2017) 34–42.
[8] B. Jan, H. Farman, M. Khan, M. Talha, I.U. Din, Designing a smart transportation system: An internet of things and big data approach, IEEE Wirel. Commun. 26 (4) (2019) 73–79.
[9] K. Darshan, K. Anandakumar, A comprehensive review on usage of Internet of Things (IoT) in healthcare system, in: 2015 International Conference on Emerging Research in Electronics, Computer Science and Technology, ICERECT, IEEE, 2015, pp. 132–136.
[10] N. Shariatzadeh, T. Lundholm, L. Lindberg, G. Sivard, Integration of digital factory with smart factory based on Internet of Things, Procedia Cirp 50 (2016) 512–517.
[11] C. Tomazzoli, S. Scannapieco, M. Cristani, Internet of things and artificial intelligence enable energy efficiency, J. Ambient Intell. Humaniz. Comput. 14 (5) (2023) 4933–4954.
[12] M. Kassab, J. DeFranco, P. Laplante, A systematic literature review on internet of things in education: Benefits and challenges, J. Comput. Assist. Learn. 36 (2) (2020) 115–127.
[13] N. Hossein Motlagh, M. Mohammadrezaei, J. Hunt, B. Zakeri, Internet of Things (IoT) and the energy sector, Energies 13 (2) (2020) 494.
[14] Y. Yuehong, Y. Zeng, X. Chen, Y. Fan, The internet of things in healthcare: An overview, J. Ind. Inf. Integr. 1 (2016) 3–13.
[15] T.T. Dandala, V. Krishnamurthy, R. Alwan, Internet of Vehicles (IoV) for traffic management, in: 2017 International Conference on Computer, Communication and Signal Processing, ICCCSP, IEEE, 2017, pp. 1–4.
[16] H. Taslimasa, S. Dadkhah, E.C.P. Neto, P. Xiong, S. Ray, A.A. Ghorbani, Security issues in Internet of Vehicles (IoV): A comprehensive survey, Internet Things (2023) 100809.
[17] J. Contreras-Castillo, S. Zeadally, J.A. Guerrero-Ibañez, Internet of vehicles: architecture, protocols, and security, IEEE Internet Things J. 5 (5) (2017) 3701–3709.
[18] M.A. Rahman, M.A. Rahim, M.M. Rahman, N. Moustafa, I. Razzak, T. Ahmad, M.N. Patwary, A secure and intelligent framework for vehicle health monitoring exploiting big-data analytics, IEEE Trans. Intell. Transp. Syst. 23 (10) (2022) 19727–19742.
[19] P. Killeen, B. Ding, I. Kiringa, T. Yeap, IoT-based predictive maintenance for fleet management, Procedia Comput. Sci. 151 (2019) 607–613.
[20] P. Singh, T. Sethi, B.B. Biswal, S.K. Pattanayak, A smart anti-theft system for vehicle security, Int. J. Mater. Mech. Manuf. 3 (4) (2015) 249–254.
[21] H. Lu, Q. Liu, D. Tian, Y. Li, H. Kim, S. Serikawa, The cognitive internet of vehicles for autonomous driving, IEEE Netw. 33 (3) (2019) 65–73.
[22] C. Hu, W. Fan, E. Zeng, Z. Hang, F. Wang, L. Qi, M.Z.A. Bhuiyan, Digital twin-assisted real-time traffic data prediction method for 5G-enabled internet of vehicles, IEEE Trans. Ind. Inform. 18 (4) (2021) 2811–2819.

[23] J.A. Fadhil, Q.I. Sarhan, Internet of Vehicles (IoV): a survey of challenges and solutions, in: 2020 21st International Arab Conference on Information Technology, ACIT, IEEE, 2020, pp. 1–10.

[24] J. Wang, K. Zhu, E. Hossain, Green Internet of Vehicles (IoV) in the 6G era: Toward sustainable vehicular communications and networking, IEEE Trans. Green Commun. Netw. 6 (1) (2021) 391–423.

[25] Y. Lu, L. Da Xu, Internet of Things (IoT) cybersecurity research: A review of current research topics, IEEE Internet Things J. 6 (2) (2018) 2103–2115.

[26] M. Kuzlu, C. Fair, O. Guler, Role of artificial intelligence in the Internet of Things (IoT) cybersecurity, Discov. Internet Things 1 (2021) 1–14.

[27] B. Kaur, S. Dadkhah, F. Shoeleh, E.C.P. Neto, P. Xiong, S. Iqbal, P. Lamontagne, S. Ray, A.A. Ghorbani, Internet of Things (IoT) security dataset evolution: Challenges and future directions, Internet Things (2023) 100780.

[28] S.H. Javed, M.B. Ahmad, M. Asif, S.H. Almotiri, K. Masood, M.A.A. Ghamdi, An intelligent system to detect advanced persistent threats in industrial internet of things (I-IoT), Electronics 11 (5) (2022) 742.

[29] R. Rizal, I. Riadi, Y. Prayudi, Network forensics for detecting flooding attack on internet of things (IoT) device, Int. J. Cybersecur. Digit. Forensics 7 (4) (2018) 382–390.

[30] E.C.P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, A.A. Ghorbani, CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment, 2023, Preprints.

[31] Z. Shah, I. Ullah, H. Li, A. Levula, K. Khurshid, Blockchain based solutions to mitigate distributed denial of service (DDoS) attacks in the Internet of Things (IoT): A survey, Sensors 22 (3) (2022) 1094.

[32] X. Cheng, J. Zhang, Y. Tu, B. Chen, Cyber situation perception for Internet of Things systems based on zero-day attack activities recognition within advanced persistent threat, Concurr. Comput.: Pract. Exper. 34 (16) (2022) e6001.

[33] H.H.R. Sherazi, R. Iqbal, F. Ahmad, Z.A. Khan, M.H. Chaudary, DDoS attack detection: A key enabler for sustainable communication in internet of vehicles, Sustain. Comput.: Inform. Syst. 23 (2019) 13–20.

[34] C.-M. Chen, B. Xiang, Y. Liu, K.-H. Wang, A secure authentication protocol for internet of vehicles, Ieee Access 7 (2019) 12047–12057.

[35] J. Ye, L. Guo, B. Yang, F. Li, L. Du, L. Guan, W. Song, Cyber–physical security of powertrain systems in modern electric vehicles: Vulnerabilities, challenges, and future visions, IEEE J. Emerg. Sel. Top. Power Electron. 9 (4) (2020) 4639–4657.

[36] Y. Sun, L. Wu, S. Wu, S. Li, T. Zhang, L. Zhang, J. Xu, Y. Xiong, Security and privacy in the Internet of Vehicles, in: 2015 International Conference on Identification, Information, and Knowledge in the Internet of Things, IIKI, IEEE, 2015, pp. 116–121.

[37] A. Sivanathan, H.H. Gharakheili, V. Sivaraman, Managing IoT cyber-security using programmable telemetry and machine learning, IEEE Trans. Netw. Serv. Manag. 17 (1) (2020) 60–74.

[38] M. Suresh, R. Anitha, Evaluating machine learning algorithms for detecting DDoS attacks, in: Advances in Network Security and Applications: 4th International Conference, CNSA 2011, Chennai, India, July 15-17, 2011 4, Springer, 2011, pp. 441–452.

[39] M. Zekri, S. El Kafhali, N. Aboutabit, Y. Saadi, DDoS attack detection using machine learning techniques in cloud computing environments, in: 2017 3rd International Conference of Cloud Computing Technologies and Applications, CloudTech, IEEE, 2017, pp. 1–7.

[40] I. Galanis, P. Gurunathan, D. Burkard, I. Anagnostopoulos, Weather-based road condition estimation in the era of Internet-of-Vehicles (IoV), in: 2018 IEEE International Symposium on Circuits and Systems, ISCAS, IEEE, 2018, pp. 1–5.

[41] F. Martinelli, F. Mercaldo, V. Nardone, A. Santone, Car hacking identification through fuzzy logic algorithms, in: 2017 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE, IEEE, 2017, pp. 1–7.

[42] X. Wang, Z. Ning, X. Hu, L. Wang, B. Hu, J. Cheng, V.C. Leung, Optimizing content dissemination for real-time traffic management in large-scale internet of vehicle systems, IEEE Trans. Veh. Technol. 68 (2) (2018) 1093–1105.

[43] Z. Khan, A. Koubaa, H. Farman, Smart route: Internet-of-vehicles (iov)-based congestion detection and avoidance (iov-based cda) using rerouting planning, Appl. Sci. 10 (13) (2020) 4541.

[44] W.-J. Chang, L.-B. Chen, K.-Y. Su, DeepCrash: A deep learning-based Internet of vehicles system for head-on and single-vehicle accident detection with emergency notification, IEEE Access 7 (2019) 148163–148175.

[45] H. Cheng, J. Yang, M. Shojafar, J. Cao, N. Jiang, Y. Liu, VFAS: Reliable and privacy-preserving V2F authentication scheme for road condition monitoring system in IoV, IEEE Trans. Veh. Technol. (2023).

[46] M. Kostrzewski, M. Marczewska, L. Uden, The Internet of Vehicles and sustainability—Reflections on environmental, social, and corporate governance, Energies 16 (7) (2023) 3208.

[47] S.A. Elsagheer Mohamed, K.A. AlShalfan, Intelligent traffic management system based on the internet of vehicles (IoV), J. Adv. Transp. 2021 (2021) 1–23.

[48] E.L. Thompson, A.G. Taye, W. Guo, P. Wei, M. Quinones, I. Ahmed, G. Biswas, J. Quattrociocchi, S. Carr, U. Topcu, et al., A survey of eVTOL aircraft and AAM operation hazards, in: AIAA AVIATION 2022 Forum, 2022, p. 3539.

[49] D. Singh, G. Tripathi, S.C. Shah, R. da Rosa Righi, Cyber physical surveillance system for Internet of Vehicles, in: 2018 IEEE 4th World Forum on Internet of Things, WF-IoT, IEEE, 2018, pp. 546–551.

[50] F. Farivar, M.S. Haghighi, A. Jolfaei, S. Wen, On the security of networked control systems in smart vehicle and its adaptive cruise control, IEEE Trans. Intell. Transp. Syst. 22 (6) (2021) 3824–3831.

[51] S.A. Elsagheer Mohamed, K.A. Alshalfan, M.A. Al-Hagery, M.T. Ben Othman, Safe driving distance and speed for collision avoidance in connected vehicles, Sensors 22 (18) (2022) 7051.

[52] W. Zhao, S. Gong, D. Zhao, F. Liu, N. Sze, H. Huang, Effects of collision warning characteristics on driving behaviors and safety in connected vehicle environments, Accid. Anal. Prev. 186 (2023) 107053.

[53] Y. Dong, Z. Hu, K. Uchimura, N. Murayama, Driver inattention monitoring system for intelligent vehicles: A review, IEEE Trans. Intell. Transp. Syst. 12 (2) (2010) 596–614.

[54] G. Raja, P. Dhanasekaran, S. Anbalagan, A. Ganapathisubramaniyan, A.K. Bashir, SDN-enabled traffic alert system for IoV in smart cities, in: IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS, IEEE, 2020, pp. 1093–1098.

[55] P. Falcone, F. Borrelli, J. Asgari, H.E. Tseng, D. Hrovat, Predictive active steering control for autonomous vehicle systems, IEEE Trans. Control Syst. Technol. 15 (3) (2007) 566–580.

[56] J. Guo, P. Hu, R. Wang, Nonlinear coordinated steering and braking control of vision-based autonomous vehicles in emergency obstacle avoidance, IEEE Trans. Intell. Transp. Syst. 17 (11) (2016) 3230–3240.

[57] S. Ponnan, S. Shelly, M.Z. Hussain, M. Ashraf, A. Haldorai, et al., Autonomous navigation system based on a dynamic access control architecture for the internet of vehicles, Comput. Electr. Eng. 101 (2022) 108037.

[58] J. Ge, Y. Luo, G. Tei, Real-time pedestrian detection and tracking at nighttime for driver-assistance systems, IEEE Trans. Intell. Transp. Syst. 10 (2) (2009) 283–298.

[59] B. Thunyapoo, C. Ratchadakorntham, P. Siricharoen, W. Susutti, Self-parking car simulation using reinforcement learning approach for moderate complexity parking scenario, in: 2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, ECTI-CON, IEEE, 2020, pp. 576–579.

[60] Z. Liu, H. Jia, R. Wu, J. Tian, G. Wang, IoV-Based mathematic model for platoon give way to emergency vehicles promptly, IEEE Trans. Intell. Transp. Syst. 23 (9) (2022) 16280–16289.

[61] A. Thakur, R. Malekian, Internet of vehicles communication technologies for traffic management and road safety applications, Wirel. Pers. Commun. 109 (2019) 31–49.
[62] B. HomChaudhuri, A. Vahidi, P. Pisu, Fast model predictive control-based fuel efficient control strategy for a group of connected vehicles in urban road conditions, IEEE Trans. Control Syst. Technol. 25 (2) (2016) 760–767.
[63] Q. Zhou, Z. Yang, K. Zhang, K. Zheng, J. Liu, A decentralized car-sharing control scheme based on smart contract in internet-of-vehicles, in: 2020 IEEE 91st Vehicular Technology Conference, VTC2020-Spring, IEEE, 2020, pp. 1–5.
[64] J.F. González-Saavedra, M. Figueroa, S. Céspedes, S. Montejo-Sánchez, Survey of cooperative advanced driver assistance systems: from a holistic and systemic vision, Sensors 22 (8) (2022) 3040.
[65] A. Gupta, A. Anpalagan, L. Guan, A.S. Khwaja, Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues, Array 10 (2021) 100057.
[66] F. Marino, F. Leccese, S. Pizzuti, Adaptive street lighting predictive control, Energy Procedia 111 (2017) 790–799.
[67] O. Maksimychev, E. Matiukhina, A. Ostroukh, Y. Vasiliev, Connected vehicle remote diagnostic system, in: 2021 Systems of Signals Generating and Processing in the Field of on Board Communications, IEEE, 2021, pp. 1–5.
[68] S. Liu, C. Wu, J. Huang, Y. Zhang, M. Ye, Y. Huang, Blockchain-based interpretable electric vehicle battery life prediction in IoV, IEEE Internet Things J. (2023).
[69] A. Petrillo, A. Picariello, S. Santini, B. Scarciello, G. Sperli, Model-based vehicular prognostics framework using Big Data architecture, Comput. Ind. 115 (2020) 103177.
[70] M. Bagheri, M. Siekkinen, J.K. Nurminen, Cellular-based vehicle to pedestrian (V2P) adaptive communication for collision avoidance, in: 2014 International Conference on Connected Vehicles and Expo, ICCVE, IEEE, 2014, pp. 450–456.
[71] E. Seo, H.M. Song, H.K. Kim, GIDS: GAN based intrusion detection system for in-vehicle network, in: 2018 16th Annual Conference on Privacy, Security and Trust, PST, 2018, pp. 1–6, http://dx.doi.org/10.1109/PST.2018.8514157.
[72] H. Lee, S.H. Jeong, H.K. Kim, OTIDS: A novel intrusion detection system for in-vehicle network by using remote frame, in: 2017 15th Annual Conference on Privacy, Security and Trust, PST, 00, 2017, pp. 57–5709, http://dx.doi.org/10.1109/PST.2017.00017, URL http://doi.ieeecomputersociety.org/10.1109/PST.2017.00017.
[73] M.E. Verma, M.D. Iannacone, R.A. Bridges, S.C. Hollifield, B. Kay, F.L. Combs, Road: The real ornl automotive dynamometer controller area network intrusion detection dataset (with a comprehensive can ids dataset survey & guide), 2020, arXiv preprint arXiv:2012.14600.
[74] I. Sharafaldin, A.H. Lashkari, A.A. Ghorbani, Toward generating a new intrusion detection dataset and intrusion traffic characterization, ICISSp 1 (2018) 108–116.
[75] A. Shiravi, H. Shiravi, M. Tavallaee, A.A. Ghorbani, Intrusion detection evaluation dataset (ISCXIDS2012), 2018.
[76] S. Almutlaq, A. Derhab, M.M. Hassan, K. Kaur, Two-stage intrusion detection system in intelligent transportation systems using rule extraction methods from deep neural networks, IEEE Trans. Intell. Transp. Syst. (2022).
[77] L. Yang, A. Shami, A transfer learning and optimized CNN based intrusion detection system for Internet of Vehicles, 2022, arXiv preprint arXiv:2201.11812.
[78] L. Yang, A. Moubayed, A. Shami, MTH-IDS: a multitiered hybrid intrusion detection system for Internet of vehicles, IEEE Internet Things J. 9 (1) (2021) 616–632.
[79] R. Kumar, P. Kumar, R. Tripathi, G.P. Gupta, N. Kumar, M.M. Hassan, A privacy-preserving-based secure framework using blockchain-enabled deep-learning in cooperative intelligent transport system, IEEE Trans. Intell. Transp. Syst. (2021).
[80] A. Rosay, F. Carlier, P. Leroux, Feed-forward neural network for Network Intrusion Detection, in: 2020 IEEE 91st Vehicular Technology Conference, VTC2020-Spring, IEEE, 2020, pp. 1–6.
[81] L. Yang, A. Moubayed, I. Hamieh, A. Shami, Tree-based intelligent intrusion detection system in internet of vehicles, in: 2019 IEEE Global Communications Conference, GLOBECOM, IEEE, 2019, pp. 1–6.
[82] M. Tavallaee, E. Bagheri, W. Lu, A.A. Ghorbani, A detailed analysis of the KDD CUP 99 data set, in: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ieee, 2009, pp. 1–6.
[83] N. Moustafa, J. Slay, UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set), in: 2015 Military Communications and Information Systems Conference, MilCIS, IEEE, 2015, pp. 1–6.
[84] N. Moustafa, A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets, Sustainable Cities Soc. 72 (2021) 102994.
[85] D. Dua, C. Graff, UCI Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences, 2017, URL http://archive.ics.uci.edu/ml.
[86] I. Ullah, Q.H. Mahmoud, A technique for generating a botnet dataset for anomalous activity detection in IoT networks, in: 2020 IEEE International Conference on Systems, Man, and Cybernetics, SMC, IEEE, 2020, pp. 134–140.
[87] W. Lee, S.J. Stolfo, A framework for constructing features and models for intrusion detection systems, ACM Trans. Inf. Syst. Secur. (TiSSEC) 3 (4) (2000) 227–261.
[88] E. Chatzoglou, G. Kambourakis, C. Kolias, Empirical evaluation of attacks against IEEE 802.11 enterprise networks: The AWID3 dataset, IEEE Access 9 (2021) 34188–34205, http://dx.doi.org/10.1109/ACCESS.2021.3061609.
[89] J. Wang, J. Liu, N. Kato, Networking and communications in autonomous driving: A survey, IEEE Commun. Surv. Tutor. 21 (2) (2018) 1243–1274.
[90] R.B. Gmbh, CAN specification version 2.0, 1991, (Accessed January 2024), Available on http://esd.cs.ucr.edu/webres/can20.pdf.
[91] G.M. Smith, What is CAN bus (Controller Area Network) and how it compares to other vehicle bus networks, 2021, (Accessed January 2024), Available on https://dewesoft.com/daq/what-is-can-bus.
[92] M. Bozdal, M. Samie, I. Jennions, A survey on can bus protocol: Attacks, challenges, and potential solutions, in: 2018 International Conference on Computing, Electronics & Communications Engineering, ICCECE, IEEE, 2018, pp. 201–205.
[93] E. Aliwa, O. Rana, C. Perera, P. Burnap, Cyberattacks and countermeasures for in-vehicle networks, ACM Comput. Surv. 54 (1) (2021) 1–37.
[94] U. Lamping, E. Warnicke, Wireshark user's guide, Interface 4 (6) (2004) 1.
[95] A. Derhab, M. Belaoued, I. Mohiuddin, F. Kurniawan, M.K. Khan, Histogram-based intrusion detection and filtering framework for secure and safe in-vehicle networks, IEEE Trans. Intell. Transp. Syst. 23 (3) (2021) 2366–2379.
[96] S.-F. Lokman, A.T. Othman, M.-H. Abu-Bakar, Intrusion detection system for automotive Controller Area Network (CAN) bus system: a review, EURASIP J. Wireless Commun. Networking 2019 (1) (2019) 1–17.
[97] T. Huang, J. Zhou, A. Bytes, ATG: An attack traffic generation tool for security testing of in-vehicle CAN bus, in: Proceedings of the 13th International Conference on Availability, Reliability and Security, 2018, pp. 1–6.
[98] GIAC, Global information assurance certification paper, 2003.
[99] Ford Mondeo CAN bus hacking, 2024, (Accessed January 2024), Available on http://www.electronicsworkshop.eu/FordMondeoCANhacking.
[100] R. Bapat, A. Mandya, X. Liu, B. Abraham, D.E. Brown, H. Kang, M. Veeraraghavan, Identifying malicious botnet traffic using logistic regression, in: 2018 Systems and Information Engineering Design Symposium, SIEDS, IEEE, 2018, pp. 266–271.
[101] M. Choubisa, R. Doshi, N. Khatri, K.K. Hiran, A simple and robust approach of random forest for intrusion detection system in cyber security, in: 2022 International Conference on IoT and Blockchain Technology, ICIBT, IEEE, 2022, pp. 1–5.

[102] M. Robnik-Šikonja, Improving random forests, in: Machine Learning: ECML 2004: 15th European Conference on Machine Learning, Pisa, Italy, September 20-24, 2004. Proceedings 15, Springer, 2004, pp. 359–370.
[103] B.M.M. AlShahrani, et al., Classification of cyber-attack using Adaboost regression classifier and securing the network, Turk. J. Comput. Math. Educ. (TURCOMAT) 12 (10) (2021) 1215–1223.
[104] A. Rehman Javed, Z. Jalil, S. Atif Moqurrab, S. Abbas, X. Liu, Ensemble adaboost classifier for accurate and fast detection of botnet attacks in connected vehicles, Trans. Emerg. Telecommun. Technol. 33 (10) (2022) e4088.
[105] F. Khan, J. Ahamed, S. Kadry, L.K. Ramasamy, Detecting malicious URLs using binary classification through ada boost algorithm, Int. J. Electr. Comput. Eng. (2088-8708) 10 (1) (2020).
[106] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, C. Wang, Machine learning and deep learning methods for cybersecurity, Ieee Access 6 (2018) 35365–35381.