Computer Communications 180 (2021) 1–20

Contents lists available at ScienceDirect

Computer Communications

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

Review

Security on in-vehicle communication protocols: Issues, challenges, and future research directions

Alfonso Martínez-Cruz [a,*], Kelsey A. Ramírez-Gutiérrez [a,*], Claudia Feregrino-Uribe [b], Alicia Morales-Reyes [b]

[a] CONACyTInstituto Nacional de Astrofísica, Óptica y Electrónica, Puebla, Mexico
[b] Computer Department, Instituto Nacional de Astrofísica, Óptica y Electrónica, Puebla, Mexico

ARTICLE INFO

Keywords:
Automotive protocols
Security
Hardware architecture
In-vehicle communication

ABSTRACT

The automotive industry has represented an important sector of industrial development worldwide. With technology growth, vehicles have been equipped with various devices that allow them to perform different functions, increase autonomy level, and give drivers greater confidence and comfort. In this sense, data security is a fundamental feature since access to car functionalities needs to be protected from outsiders, only allowing access to authorized users. It means to guarantee data security through communication buses even when connecting from external devices. Therefore, it is necessary to design new security techniques that increase the confidentiality and improve authentication in new vehicles.

Although at first, automotive communication protocols did not include security mechanisms, the risk of threats was always latent. Due to this, the search for new methods and architectures to improve security and autonomous functions in communication protocols has been in constant increase. New paradigms and ways to prevent, detect and mitigate attacks on automotive communication are necessary for the new challenges ahead. This article comparatively analyses the state of the art for in-vehicle communications protocols regarding a variety of algorithmic approaches based on architectural configurations and extensively compares their performance.

Contents

* Corresponding authors.
E-mail addresses: amartinez@inaoep.mx (A. Martínez-Cruz), kramirez@inaoep.mx (K.A. Ramírez-Gutiérrez), cferegrino@inaoep.mx (C. Feregrino-Uribe), a.morales@inaoep.mx (A. Morales-Reyes).

https://doi.org/10.1016/j.comcom.2021.08.027
Received 12 February 2021; Received in revised form 21 July 2021; Accepted 30 August 2021
Available online 3 September 2021

1. Introduction

The arrival of new technologies and more optimized communication systems has led to the creation of smart-cars, with greater functionality and better levels of autonomy. Moreover, to achieve more efficient communication systems between different devices, new communication protocols have been designed. A communication protocol can be defined as a set of rules that allow two or more entities to communicate within a communication system and send information through a channel or a physical medium.

The future of new car design is a fast moving open topic because of recent technological advances such as sensor technology, IoT, 3D printing, and Industry 4.0. Automotive industries, as well as digital application services companies, have to adapt to constant changes caused by those technological advances.

With increased functionality and connectivity, more complex lines of code have been included in cars. According to experts, the number of code lines in cars surpasses 100 million, so far exceeding the code in common airplanes and navigation systems [1].

Currently, in automobiles, most mechanical components have been replaced by electromechanical systems. Each time, more sensors and electronic devices are included in automotive communication networks. Because of this, a large number of wires are required to connect different devices, so manufacturers have designed different types of bus systems to reduce the number of wires, but most of these buses are incompatible with each other. Also, current trends are connected cars and autonomous driving, in which cars are cyber–physical systems that interact with the environment and control different variables based on digitalization [2].

To provide services, functions, and applications required in cars, there are different communication protocols used by the automotive industry, such as: CAN, MOST, FlexRay, LIN, Ethernet, etc [3].

Although currently there is no single criterion for using a unique protocol, different companies have used CAN to improve information handling and applications developed for cars.

Advance in automotive communications and connectivity has attracted attention and generated attacks from unauthorized users. Now, cyberattacks can be more frequently found in digital communications, and in the specific case of automobiles, it has been shown that communication buses are not isolated from the problem, in fact, hackers have shown that automobiles can be attacked and controlled by taking advantage of such security vulnerabilities. Due to this, in the last years, more research works focused on increasing security on the in-vehicle network have been proposed. Fig. 1 shows published articles related to the security schemes proposed on in-vehicle protocols between 2014 and 2020. In this graph, the proposals in methods based on Cryptography, Machine Learning, Neural Networks, among others, have been classified. As it can be seen, there has been a considerable increase in the last three years, where methods based on other alternatives have also been proposed.

Nowadays, with new features such as telematics and Advanced Driver Assistance Systems (ADAS), experts mention that a single modern luxury vehicle now can integrate as many as 150 Electronic Control Units (ECUs). Every ECU directs a specific function in the vehicle, for example, body control, brakes control, seat control, smart antenna module, door/window control, among others. There are different electronic components within cars, some of them are shown Fig. 2, such as front unit screen, rear camera and light control unit.

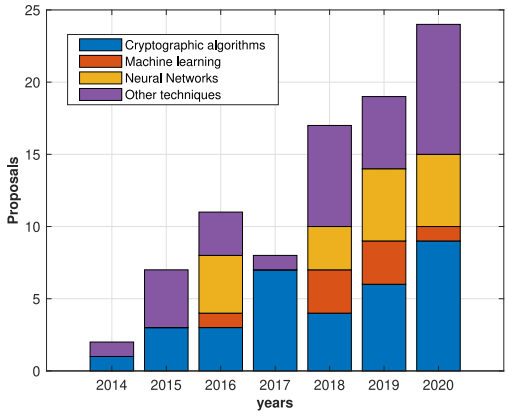

Fig. 1. Algorithmic approaches published for Security on in-vehicle protocols.

Fig. 2. Electronic components in a car [4].

With the increase in the number of ECUs, a global association founded in 2013 by entities interested in the automotive industry (automobile manufacturers, electronics and software providers, and tool providers) developed AUTOSAR (AUTomotive Open System ARchitecture), the global standard for automotive software architecture. Its main objective is to create and establish an standard and open software architecture for ECUs as well as including scalability, software transferability, consideration of availability and security requirements.

At first, communication protocols in the automotive systems were not designed to face up cyberattacks, as is the case of the CAN bus, which was not originally developed to protect data frames. However, now, cyberattacks are increasingly recurrent and, therefore, it is necessary to propose new systems that guarantee data security and ensure proper systems performance.

Industry experts have suggested some requirements for the proposed solutions to be considered viable [5]. Among the most relevant are the following:

- Profitability.
- Compatibility with previous versions.
- Support for car repair.
- A detailed description of the implementation.
- Reasonable overhead.

Also, some aspects to be considered and are involved in the safety of the vehicle are:

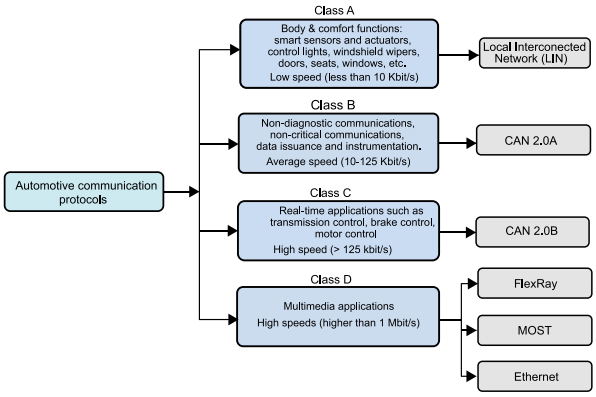

Fig. 3. Automotive protocols classes.

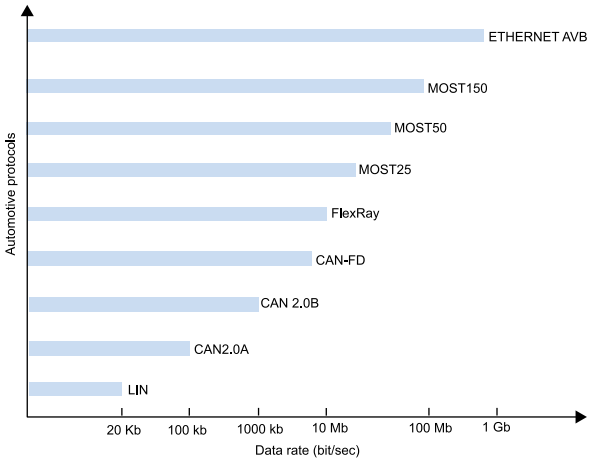

Fig. 4. Main automotive protocols data rates.

- Hardware and software protection for CAN bus and ECUs
- Defense techniques based on software V2V, network monitoring, and communication in and out of the car.
- V2X security, data integrity, and privacy.

Protocols have been the core of communications. According to the Society of Automotive Engineers (SAE), communication protocols are split into four different classes depending on the application , for example, Class A: LIN, Class B: CAN 2.0 A, Class C: CAN 2.0B and FlexRay and MOST in class D, as shown in Fig. 3.

According to data transmission, communication protocols can be designed based on two different paradigms:

- Event Triggered. This is performed based on events that generate spontaneous messages.
- Time Triggered. Activities are triggered by time progression.

In protocols based on activated events, frame transmission is performed according to certain events generated during communication between devices. Unlike the previous case, in the protocols triggered by time under the time activation scheme, the transmission is carried out following the time intervals, which are generally configured at the beginning of communication.

With the increase of bit rates in communication protocols, new applications emerge and, in turn, with the connection of different buses, more complex networks and greater robustness in the car's functionality are supported. Fig. 4 shows a data rate comparison of the main automotive protocols.

In this paper, a current overview on cybersecurity techniques, issues and tools following a taxonomy based on the security architectural type and associated algorithmic techniques is presented. This survey is organized as follows: In Section 2, surveys on security technologies and contributions are shown. In Section 3, a discussion on threats and challenges on in-vehicle protocols is presented. Section 4 presents Security solutions on CAN bus. In Section 5 a review of other automotive protocols such as LIN, FlexRay, MOST, Ethernet is presented. Finally, in Section 7 conclusions are drawn.

2. Surveys on security technologies and contributions

In recent years, the research community has paid increased attention to new security schemes and intrusion detection systems in automotive protocols aiming at the CAN bus mainly. Thus, a number of related surveys targeting security aspects on in-vehicle protocols have been recently published. Table 1 shows closely related surveys that from different perspectives analyze security technologies and techniques on in-vehicle protocols.

In [5], a review on promising CAN message authentication solutions is presented together with an analysis of main requirements needed for adoption by the industry. Later in [6], a survey on attacks and solutions targeting the CAN bus is drawn by analyzing its security vulnerabilities due to the lack of encryption and authentication mechanisms.

Fast technological development in the vehicle industry has mainly relied on the CAN bus without carefully consider security aspects. In [7] a survey on a decade of proposed solutions to improve CAN security is introduced with a remark on their maturity and readiness for industry. A number of software applications have been developed to increase users safety and comfort, such benefits also poses new security challenges. Huynh et al. reviewed related protection mechanisms for automotive applications in order to identify those opportunity gaps for improvement [8].

Intrusion Detection Systems (IDSs) have been one of the main approaches to protect the CAN bus from attacks. In [9] a very complete survey on those IDS proposals is presented. Following IDS review, Lokman et al. presented an extended analysis on IDS approaches considering their basic concepts such as frequency, statistics, machine learning or hybrid based criterion [10] . Another survey on security mechanisms considering cryptography and IDS was presented in [11]. Gmiden et al. discussed related research works that approach security issues on the CAN bus and also introduce an IDS to address them.

A three-layer security framework to deal with a number of attacks at sensing, communications and control fronts is presented in [13]. In that review, interaction effects among those aspects are analyzed from an attack perspective with remarks on security mechanisms to increase user reliability in modern vehicles.

Different to previously discussed articles, this paper analyses the state of the art for security on in-vehicle communications protocols through a taxonomy based on the security architecture type considering three main ones, see Fig. 7: (a) central type where security solutions are managed through an specific in-vehicle network's module, (b) semi-distributed type,where security schemes involve a number of nodes within the network; and (c) distributed type, where security mechanisms are ubiquitous to all nodes within the network. Once architecture types are determined, associated algorithmic techniques are analyzed accordingly. It is a contribution in this work to present those algorithmic mechanisms that have achieved high performance metrics and those still in pursue of improving the state of the art but offer some advantages to the current development in the security area for in-vehicle communications protocols.

3. Automotive security issues and solutions

Over the years, several works about cyberattacks in Automotive Communication Protocols have been proposed, [14–16]. According to these works, attacks on in-vehicle networks were achieved [17]. One of the attacks that attracted the most attention was published in 2015 [18,

Table 1
Security solutions on in-vehicle protocols related surveys.

Year	Author	Contributions
2017	Nowdehi, et al. [5]	Analysis on CAN message authentication solutions and security solutions requirements for implementation.
2018	Bozdal, et al. [6]	Vulnerabilities description and related work on attacks and solutions for CAN bus.
2018	Groza, et al. [7]	Security solutions based on cryptography for the CAN bus.
2018	Huynh, et al. [8]	Review security and privacy in automotive application platforms.
2019	Young, et al. [9]	Survey on Intrusion Detection Systems (IDS) for CAN bus.
2019	Lokman, et al. [10]	Survey on Intrusion Detection Systems (IDS) for CAN bus based on detection approaches and techniques for attacks and challenges.
2019	Gmiden, et al. [11]	Several approaches to improve CAN bus security and a tool to evaluate cryptographic algorithms.
2020	Wu, et al. [12]	Main works on Intrusion Detection Systems (IDS) and attacks on in-vehicle Networks based on techniques, technologies and their contribution.
2020	El-Rewini, et al. [13]	Cyber attacks in automotive and V2x protocols with countermeasures. In addition, a three-layer classification scheme: control, communication, and sensing to carry out cyber attacks.

19], it was performed on a Jeep Grand Cherokee, hackers took remote control of the vehicle by accessing the infotainment system.

According to Upstream [20], attacks on cars can be classified into physical and remote, in the first one the attacker needs to be physically connected to the vehicle whereas in the second one no connection is required, these can be performed in a short or long-range. Remote attacks have increased lately, representing 79.6% of the registered attacks, which generally rely on network connectivity (like radio transmissions, Wi-Fi, Bluetooth, 3/4/5G networks, etc.). In early 2020, Keen Team researchers identified vulnerabilities in the Mercedes-Benz User Experience (MBUX), the infotainment system initially introduced on A-class vehicles in 2018, they found an outdated Linux kernel that was susceptible to specific attacks, and sent specific CAN messages, being able to control the ambient and the reading lights, open the sunshade cover and control the back-seat passenger lights, but they could not take control of the vehicle [21]. In February 2016, Nissan disabled their Leaf mobile app due to cyber security vulnerabilities that allowed to control of climate and access to location data [22]. Man-in-the-Middle attacks in Wi-Fi connection were performed to Hyundai's BlueLink mobile app exposing personal data and vehicle controls to intruders [23].

The risk of cyber attacks is clear, it is not only about personal data or cars being stolen but about human lives exposed to danger. It has been showed that attackers can take control of cars by accessing from different scenarios, and if wanted this could provoke catastrophic accidents.

An important security challenge present on in-vehicle communications is the Intrusion Detection System (IDS), mentioned earlier, which can be implemented in hardware or software to automatically detect attacks. Different IDS types exist, like Host-based (HIDS) and Network-based (NIDS); these are an elements in a system for network traffic monitoring. There are different research works where IDS and attacks on in-vehicle networks have been studied [10,12,24], and others focused on mechanisms to detect intruders in the CAN bus, for example in [24], authors analyze vulnerabilities and propose a design for a real-time intrusion prevention system (IPS) that counteracts attacks by monitoring the Control Area Network bus and eliminates malicious messages.

Security on in-vehicle networks can be analyzed based on protocols and techniques designed for different automotive components involved in them, for example, Engine Control Module, Transmission control module, Electronic Brake Control Module, Body and Telematics modules; in these different attack surfaces are involved: OBD-II, Bluetooth, Wi-Fi, Keyless etc. Table 2 shows the main automotive attack surfaces matching both attacks and countermeasures.

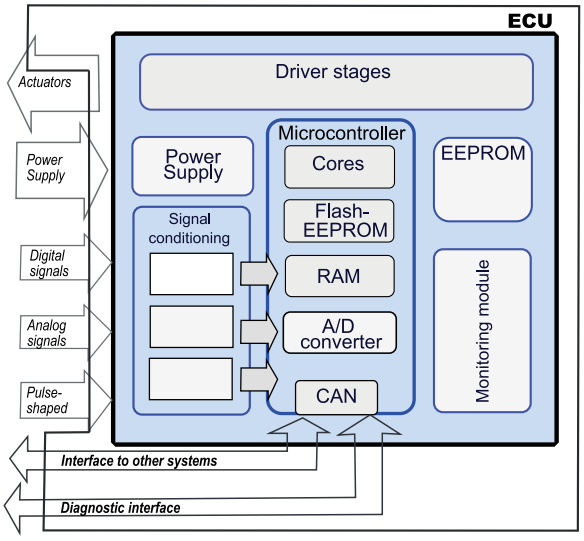

Fig. 5. Main Electronic Control Unit (ECU) components.

In remote keyless systems, some works to improve security have been proposed [25,26]. For example, in [27], authors proposed a security mechanism based on timestamping and XOR encoding to mitigate replay and jamming attacks in RKE. In their experiments, they implemented a prototype with an AES key compromised. They show that it is feasible to use their mechanism to mitigate generated replay and interference attacks.

Protecting vehicle functions from access and tampering with unauthorized individuals is a primary challenge for ECU developers. As mentioned, an ECU is an embedded system responsible for processing signals received from several sensors to measure different variables (temperature, air, speed, etc.). Depending on its operation, there are different types of ECUs, like powertrain, vehicle safety, comfort, infotainment, engine control, telematics, etc. They are placed in different parts of the car and control several important units. It means, that an ECU is a microcomputer system that generally has a microcontroller, a specific signal processing unit, and a controller IC, as well as electronic components and power supply, among others.

Table 2
Attacks and countermeasures on different attack surfaces on in-vehicle networks.

Access control ports	Works	Attacks	Countermeasures
OBD-II	[28,29]	Injection attack, DoS, spoofing messages, man in the middle attack, Fuzzy Attack, replay attack, masquerade attack	Message authentication between devices, key management, embed the malware, establishing session keys, encrypt messages, cryptographic algorithms, PUFs.
Bluetooth	[30–32]	Surveillance, Fuzzing, Denial of Service (DoS), Malware, Man in the middle, eavesdropping, Sniffing, brute force, buffer overflow, Backdoor Attack.	Cryptographic algorithms, ensure device mutual authentication, configure encryption key sizes to the maximum allowable.
Wi-Fi	[19,33,34]	Shell injection, malicious smartphone app, DoS attacks, spoofing attacks	Security protocols, encryption and hash algorithms, filter incoming packets, blocking the download and installation of malicious APK files.
LiDAR	[35–37]	Tampering attacks, noise-addition attacks, Sensor saturation, spoofing attacks, interference.	Data integrity verification, tamper detection techniques, correlation-based detection, pulse detection, beat frequency detection, redundancy.
CD	[29,30]	Malicious code, firmware update.	Malicious software detection.
Radar	[38–40]	spoofing attacks, DoS attacks, injection attacks, Jamming attacks	Sensor fusion with LiDAR/camera, phase randomization, frequency randomization.
Keyless Entry Systems	[41–43]	Relay attacks, jamming, replay attacks, scan attacks, playback attacks, forward prediction, dictionary attack, two-thief attack, interferences	Shielding the key, removing the battery from the key, hardware modification, security protocols, increase the number of challenge bits, changes of transmitted code, cryptographic algorithms.

Sensors are fundamental, they acquire signals from different parts of the vehicle and deliver it to the ECU, which is responsible for their processing. In turn, it processes data to send control signals to actuators to perform some function. At a software level, the ECU running program is stored in RAM memory and is usually executed by a microcontroller, which is the ECU's core. Therefore, algorithms performing different closed-loop control functions are processed by this device [44]. Fig. 5 shows the main components in an Electronic Control Unit.

ECUs should be designed to function properly under different conditions, considering for example, extreme temperature ranges, exposure to fluids (oil, fuel, etc.), mechanical stress (vibrations), exposure to moisture, among others.

Today, ECUs data security is very important, and should start at hardware level where a secure boot is guaranteed. Moreover, software protection allows flexibility in the implementation of cryptographic algorithms. In [45], authors proposed an MCU and FPGA co-designed ECU architecture for automotive systems, they showed their system communicates in an ECU environment. However, a robust security scheme is required.

An architecture scheme for ECUs with security schemes is proposed in [46]. Authors use an FPGA ECU implementation. The proposal integrates confidentiality, integrity, and authentication through AES-128 (128-bit) encryption and an SHA-3 algorithm based on HMAC. Good results are achieved in timing analysis, energy analysis, QoS and behavioral reliability, and feasibility analysis.

In [47], authors proposed a vehicular hardware security module (HSM) that enables holistic protection of in-vehicle ECUs and their communications. Their architecture contains an asymmetric and a symmetric crypto engine that enables creation and verification of digital signatures and symmetric encryption and decryption, respectively.

Currently, multiple ECUs within cars are connected through buses with different communication speeds depending on the application type and connected devices. In those communication buses, information exchange allows different functions to be implemented and more features to be integrated in smart cars. Also, a wide range of data rates are used by different protocols, from 20 kbps for the LIN Bus to 100 Mbps for wireless interfaces (G/4G/future 5G, BT, Wi-Fi, V2X). In order to interconnect data and process it, gateways have merged as a solution, allowing securely process information between different functional domains. Some of the main capabilities of a gateway are: protocol translation, data routing, diagnostic routing, firewall filtering, message mirroring, intrusion detection, network management, key management, and OTA management [48]. A security layer in the gateway should act as a firewall that controls access from the external interfaces to the vehicle's inner network, and verifies the communication between nodes [49]. Some authors have presented solutions to secure gateways in the automotive environment [50–52], where the most important is to guarantee message authentication and data integrity and privacy.

To secure automobile communications some cryptographic algorithms could be used mainly in the application layer. Among those algorithms are SHA1, SHA256, SHA3-256, HMAC-SHA1, HMAC-SHA256, HMAC-SHA3 which could be implemented on hardware with enough resources [7]. The AUTOSAR Specification of Secure Onboard Communication proposes using Message Authentication Codes (MACs) as a basis for authentication (e.g. a CMAC based on AES) [53], this in the service layer. An important challenge using cryptography is the secure storage of keys, as well as the implementation of cryptographic algorithms in a secure manner. Moreover, in order to achieve a minimum level of security, different hardware components are required. Besides, an adequate safety structure should be implemented in an automotive system.

3.1. AUTOSAR security modules

AUTOSAR has developed a specification with certain requirements applicable to the design, for example to secure on board communication, the SecOC module was developed, where the basic security features, the functionality and the API are described. On the sender side, the SecOC module produces and adds an authentication code (MAC) to every data set and optionally a freshness value is also incorporated with the result. The data, authentication, and freshness value are transmitted via CANFD within a single frame. The challenge for the security of the system is to safeguard the keys it uses [54]. On the receiver side, the **SecOC module** checks the freshness value and authenticity by verifying the authentication code that has been enclosed by the SecOC module sending side [53].

The SecOC module in addition to using MACs also uses digital signatures to guarantee that the received information comes from the right ECU and contains the correct data. To provide cryptographic functions, the module uses either the Crypto Service Manager (CSM) or the Crypto Abstraction Layer (CAL). **Crypto Stack** is an AUTOSAR

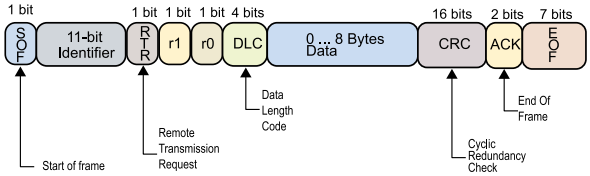

Fig. 6. CAN 2.0A frame structure.

module that offers standardized access to cryptographic services such as hashing, asymmetric signature verification, or symmetric data encryption. These services depend on cryptographic primitives and underlying cryptographic schemes. Crypto Stack allows configuring the services needed and establishing various configurations for each service where schemes and primitives can be chosen [55]. AUTOSAR crypto stack can support multiple cryptographic implementations in parallel [56].

AUTOSAR Adaptive Platform specifies the following concepts: *DTLS* for secure communication over UDP, *TLS* for secure communication over TCP, and *IPSec* for secure communication over IP [57]. Implementing **IPsec** in AUTOSAR Adaptive Platform provides options for securing communication between network nodes with confidentiality, integrity, or both. IPsec is a standard network security protocol that provides ways to secure communication while supporting multi-vendor stack interoperability [58].

Transport Layer Security (TLS/DTLS) is used to provide secure client–server communication on TCP/UDP-level. The Security Manager provides the TLS protocol stack for Ethernet communication. IPsec and TLS standards support authentic and confidential communication [56].

The **Identity and Access Management** specification provide the infrastructure to carry out access control for intra-ECU and inter-ECU operations [59]. This module assures that only authorized applications access certain resources that can be updated at any time [56].

As a result of increasing components in vehicles and connectivity, more security components like hardware modules should be implemented and incorporated into AUTOSAR as well as intrusion detection and prevention solutions, seeking to guarantee data security and ECUs functionality.

4. Security solutions on CAN bus

CAN bus is the most commonly used bus in the automotive industry. This network protocol is based on multiple masters. Its operating mechanism allows the bus to access the message without causing excessive delays. CAN protocol emerged as a solution to additional wiring costs since none of the existing protocols at that time worked efficiently in automotive applications.

CAN bus is defined as a communication protocol that is message-oriented, that is, information to be exchanged is divided in messages, each message contains an identifier and is grouped in tables for transmission. The CAN bus is a two-wire bus, its specification is in the ISO 11898 standard. The first two specification layers are defined according to physical and data link layers in the OSI model.

Fig. 6 shows details of CAN2.0 A frame structure, where the frame is composed by a start of frame, a delimiter, a remote transmission request, r0, and r1 bits, a data length code, a data section, CRC, ACK and EOF. CAN 2.0B frame structure differs from CAN2.0 A in a 18-bit identifier. This represents the extended version of the CAN bus.

Some advantages of CAN bus are:

- Minimization of the number of cables in the vehicle, which simplifies wiring and gives the car less total weight.
- Fewer sensors, since the same sensor can give the same information for several electronic systems.
- Fewer connections between control units.
- Better performance of the components.
- Measurement of all electronic aspects of the automobile.
- Integrated diagnostics and fault notification.
- Automatic retransmission of erroneous frames.
- Multicast reception with time synchronization.

4.1. Security issues

Originally, the Controller Area Network did not include security schemes that prevent attacks and abnormal conditions during communication. There are works focused on the analysis of characteristics, vulnerabilities of the CAN bus [5,6,9,60,61]. In these works, authors describe and analyze the vulnerabilities and IDSs for the CAN bus, finding the following as some of them:

- Lack of Message Authentication of the nodes. Due to each ECU delivers and receives all data on the same bus
- Unsegmented Network. Commonly, the CAN network is not segmented, due to this, components focused on safety-critical systems can communicate with infotainment in the same network.
- Lack of encrypted Messages. As the CAN bus was not designed to prevent hacker attacks, information is not encrypted, allowing access and possible modification of the data, or even its replacement and, therefore, errors in the operation of the system.
- Vulnerable to DoS attacks and replay attacks. On the CAN bus, each device receives the information that is sent from a transmitter, due to this an attacker could send frames with high priority and interrupt critical functions in the system.

Security solutions including new protocols and algorithms for the Controller Area Network or new security architectures in ECUs and gateways, and security schemes in other in-vehicle protocols have been proposed.

4.1.1. Taxonomy of security solutions

A taxonomy of recent research works about security issues and solutions in in-vehicle protocols is presented, which is based both on the type of technique proposed and on the security architecture used.

Fig. 7 shows the distribution of main works related to security technologies for CAN bus. In this case, we present a taxonomy based on the type of security architecture [62]. A centralized security architecture implies a device on the bus keeps the main management of the system, which communicates with other devices to carry out adequate security control. In a semi-distributed architecture, main security management involves a subset of devices on the bus. In a distributed security architecture, each node can take the necessary actions to protect itself and provide security on the bus.

On the other hand, there are works related security schemes on Controller Area Networks, Fig. 8 shows the proposals published between 2014 and 2020. In this case, we classify them into 4 categories: cryptography, machine learning, neural networks, and others, that in the last four years the use of these techniques has increased. It can be seen in that works that the use of schemes based on cryptography and other techniques has been proposed to increase security on in-vehicle networks.

4.2. Security schemes based on neural networks

Diverse works using Neural Networks have been proposed, for example, in [63], the authors developed an IDS through a Deep Neural Networks (DNN). In this case, network parameters were trained using probability feature vectors that were obtained from packets of an in-vehicular network. According to the experiments, the method can identify any malicious attacks on the bus and, provide a real-time response with an accurate detection ratio of 98% on average. Most of the proposed methods focus on intrusion detection schemes, where neural networks are used to detect anomalous frames on the bus and

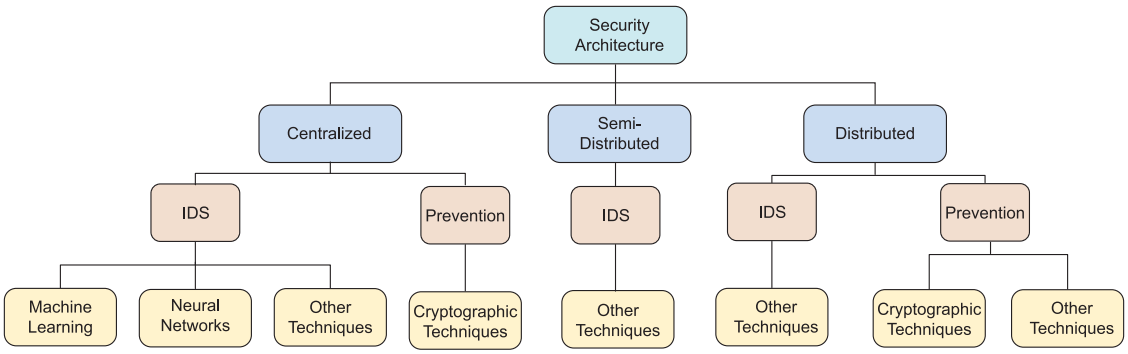

Fig. 7. Distribution of security methods for CAN bus based on security architecture type and associated algorithmic techniques.

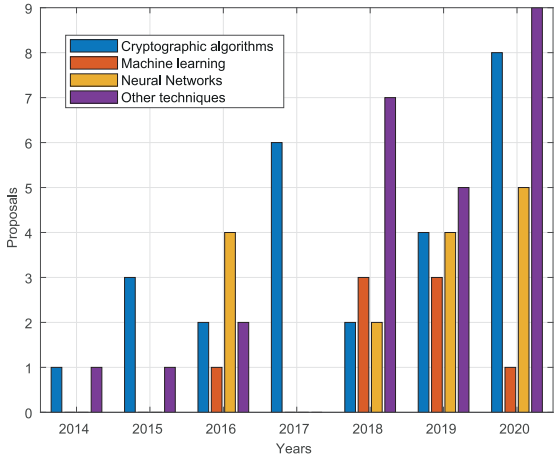

Fig. 8. Security methods on CAN bus published works.

alert the devices about the generated attacks and that the appropriate actions can be taken.

Table 3 shows the classification of the main works related to security methods using Neural Networks (NN) on CAN bus. In this case, most of the published works involve a centralized security architecture where its scheme is basically controlled by a specific device connected to the network. Although many of the published works do not mention much metrics to measure the performance of the system, we present a comparison based on the percentage of true positive rate (TPR) provided, that is, the percentage of attacks correctly detected from the attacks generated.

As it can be seen, different types of neural networks have been used to improve safety on the CAN bus. Likewise, Deep Learning and Spatio-temporal methods have been implemented with good results. A challenge is presented when implementing solutions with a broad set of attacks on hardware devices in real environments.

Among the main countermeasures proposed in these intrusion detection methods are the following: high-confidence alerts to change the state of the vehicle to safe mode to allow the driver to stop the vehicle, send alarms to analyze an anomaly, send messages to the driver and implementation of cryptographic algorithms.

4.3. Security schemes based on machine learning

Works to detect cyber attacks in CAN bus using Machine Learning have been proposed, for example, authors in [78] presented a detection model based on modified one-class support vector machine (SVM) in CAN traffic. Two main phases make up this method: training and testing or evaluation phases. Three different methods are applied to compare its performance, according to their results, this method reaches the highest hit rate and the lowest miss rate compared to other anomaly detection methods.

Table 4 shows a summary on main security methods that use Machine Learning techniques, most of them involve a centralized security architecture using Markov models and SVMs to mainly detect Denial of Service (DoS) attacks, replay, fuzzy and injection in CAN bus. Different from other methods, these works use Machine Learning techniques to detect different types of attacks on the bus through specific characteristics that lead to greater detection of anomalous data frames and increasing security in the network. According to their results, a large percentage of true positives is achieved, however, the number of cyber attacks supported varies and, in many cases, is limited. One challenge is adding mechanisms that respond and mitigate attacks and improve performance in real-world environments.

Regarding the suggested countermeasures to mitigate attacks on the proposed IDSs include the following: using cryptographic algorithms and authentication methods, implementing lightweight authentication encryption mechanisms, as well as new vehicle network architectures and implementation of bus security protocols.

4.4. Security schemes based on cryptographic algorithms

There is not a standard way to guarantee a high level of security in the CAN bus, therefore, the implementation of cryptographic algorithms has been proposed, which allows avoiding cyber attacks generated to access or alter the information contained in the data frames.

Methods to prevent unauthorized data transmission in CAN include [79–81], where different ways to increase security in CAN bus are shown.

In [82] authors investigated a lightweight CAN authentication protocol and proposed a solution for its immunity against attacks. The behavior analyzed introduces a DoS attack considered as one of the most dangerous in vehicles. Their solution consists of three stages and strengthens the weak point in a Lightweight Authentication Protocol (LCAP) in order to be more robust against such attacks.

In [83], researchers proposed an ID-Anonymization version for the CAN bus (IA-CAN). Their proposed protocol can protect against DoS attacks as well as provide a secure channel between in-vehicle components and external devices. This proposal is based on two authentication processes: the first one consists of anonymous ID filtering to check transmitter authenticity. The second one involves message authentication to perform data validation. According to their results, the protocol provides protection against DoS attacks and secure integration of external devices to the vehicle network.

Also, in [84], authors present a CAN centralized system through an improved CAN controller. In their proposal, a monitoring system using a message authentication code is proposed, in this case, a node authenticates each ECU in the bus, and then, it reviews message authentication codes, which are assigned to messages transmitted on the CAN bus. According to their proposal, one advantage is that only new

Table 3
Related works on Neural Networks solutions for detection attacks on the CAN bus.

Year	Author	Neural Network	Attacks	Contributions	TPR performance
Centralized security architecture					
2016	Min-Ju Kang et al. [63]	Deep neural network (DNN)	Packet Sniffing, Fuzzing	An intrusion detection technique that uses a deep neural network (DNN) to discriminate normal packets and attacks.	99.9–93.7%
2016	A. Taylor et al. [64]	Long Short-Term Memory neural network (Recurrent Neural Networks)	Injection attacks (suppress, Interleave, Drop)	An anomaly detector based on a Long ShortTerm Memory neural network to detect CAN bus attacks.	100–48%
2018	G. Loukas et al. [65]	A deep multilayer perceptron and recurrent neural network architecture	Denial of Service (DoS), Command injection, Malware	An Intrusion Detection Scheme based on Deep Learning to detect different type of attacks.	95.4–82.2%
2018	Wang et al. [66]	Neural Knowledge DNA	Single package, replay, flood, and complex series order attacks	Anomaly detection method and the knowledge representation of security experience for CAN bus.	98.2–80%
2019	S. Boumiza et al. [67]	Multi-Layer Perceptron (MLP)	DoS, Frequency-appearance modification	An IDS based on MLP to detect attacks on CAN bus.	100–85%
2019	H.M. Song et al [68]	Deep convolutional neural network (DCNN)	DoS Attack, Fuzzy Attack, Gear Spoofing, RPM Spoofing	An IDS based on a deep Convolutional Neural Network to protect CAN bus.	98%
2019	J. Xiao et al. [69]	spatiotemporal information/Convolutional Long short-term memory (LSTM)	DoS, Fuzzy, impersonation, attack free	Their method uses spatiotemporal information and outperforms the classical Long Short Term Memory model	100–96%
2020	M. Hanselmann et al. [70]	Deep Learning	Flooding, playback, suppress, Plateau attacks	Unsupervised method based on deep learning using long short-term memories.	93.7–75%
2020	Barletta et al. [71]	Kohonen self-organizing map	DoS, spoofing and fuzzy attacks.	A distance-based IDS to identify attack messages injected in a CAN bus using supervised and unsupervised Kohonen's Self Organizing Map Network (SOM).	97.85–67.8%

Table 4
Related works on Machine Learning solutions for detection attacks on the CAN bus.

Year	Author	ML technique	Attacks	Contribution	TPR performance
Centralized security architecture					
2016	S. N. Narayanan et al. [72]	Hidden Markov Model	Injection attacks, states attack, Random	An scheme based on Markov Model to detect malicious behaviors and deliver alerts on car operation.	-- -- --
2018	A. Alshammari et al. [73]	KNN (k-nearest Neighbor) y SVM (Support vector machine)	DoS and Fuzzy attacks	An intrusion detection mechanism (IDS) in VANET based on machine learning (KNN and SVM).	99–96%
2018	D. Tian et al. [74]	Gradient Boosting Decision Tree (GBDT)	Injection attacks (Random)	An IDS based on Gradient Boosting Decision Tree (GBDT) and entropy on CAN bus.	97.67–95%
2018	Wang, et al. [75]	Hierarchical temporal memory (HTM)	Random, replay attacks	A distributed anomaly detection system using hierarchical temporal memory (HTM) to enhance the security of CAN bus.	98%
2019	Y. Hamada et al. [76]	Divider, estimator, evaluator	Spoofing attacks(straight attack, jab attack)	Intrusion detection mechanism is based on a data estimation control (CDEC) that monitors sensor control data.	99%
2019	M. Al-Saud et al. [77]	Support Vector Machine model and social spider optimization algorithm	DoS attack	IDS based on a support Vector Machine model and a improved optimization algorithm based on social spider optimization algorithm (MSSO).	96.1%
2019	O. Avatefipour et al. [78]	SVM with MBAT algorithm	Attacked traffic log, injection attacks, DoS, attack free states	An anomaly detection model based on a modified class carrier vector machine and a bat algorithm to enforce the CAN bus network communication protocol in the vehicle.	97.01%–80%

required hardware is monitored. In the case of other ECUs, software to authenticate nodes and perform key exchange is required. They implemented their solution in hardware and, according to their results, the proposed monitoring system increases security in the CAN bus.

In [85], the authors proposed a technique that uses a truncated MAC to transmit messages and a segment of MAC in a data frame. Basically, this proposal consists of two different strategies to prevent attacks in the Controller Area Network. They implemented their method in Arduino Uno. According to the results, this protocol detects tampering and replay attacks through the use of a MAC, transaction numbers, and a key.

Other authors suggest new methods based on cryptographic techniques to increase security in CAN bus, such as [86], where researchers designed a Lightweight Encryption and Authentication Protocol (LEAP). Their method uses a stream cipher to encrypt CAN messages and a key management mechanism to protect them from external attacks. According to the results, the method consumes less memory and about 8X higher efficiency than MAC-based approaches.

Other authors have focused on improving the performance of the CAN protocol. For example, in [87], the authors proposed a technique to improve the timing behavior of encrypted CAN buses. The proposed method is to assign different priority levels to encrypted frames to reduce the delay. According to obtained results over Arduino boards and simulations, the total time is considerably reduced regarding the delay in normal data transmission.

Other works such as [88], focus on the most common causes of cyber attacks on vehicular networks, they present a Runtime Verification (RV) based framework to defend CAN protocol against malicious attacks; the framework uses a copilot to perform detection at runtime; their implementation can defend from the malicious attack before it affects a vehicular system.

Table 5 shows solutions using cryptographic techniques regarding security architectural schemes. However, many of those works do not present common metrics, therefore, most relevant aspects are summarized. Centralized and distributed architectural approaches are mainly proposed with mechanisms based on SHA-256, AES, Diffie–Hellman Elliptical Curve Cryptography, MAC, and HMAC to target CAN bus integrity and confidentiality while fulfilling compatibility criteria and timing requirements in real-world environments.

4.5. Security schemes based on other techniques

As mentioned earlier, security solutions have been explored using neural networks, machine learning, and cryptographic techniques, however, other paradigms have also been proposed. Researchers have used arrival time analysis for data frames, identifiers in data frames, mining approaches, Bloom filters, flags, reconfigurable ECUs, etc. In those solutions, a wide variety of attacks (DoS, replay, injection, spoofing, masquerade) can be detected using these techniques and competitive results are reported. Works like [97] propose an IDS using the content of the framework and its periodicity. This method is based on the detection of parts of the data field that does not change frequently, in turn, it uses Bloom filters to identify the content of the messages, as long as they are transmitted at fixed time intervals on the communication bus.

Table 6 shows the main works using different techniques, it is based on the type of technique used, the type of security architecture, and the true positive rate given. There are different solutions in centralized, semi-distributed, and distributed architectures. In the case of Distributed security architectures, most of the methods focus on ensuring security at each node using Flags and error frames, reconfigurable ECUs and, changing frame IDs mainly. In this scheme, an advantage is that all ECUs are protected increasing the safety on the bus, however, it is necessary to meet the time requirements for communication on the bus. According to the results, high detection rates based on techniques such as entropy, mining approaches, and pattern recognition have been reached.

Some of the main countermeasures proposed to mitigate the attacks detected with this type of detection system are the following: use of cryptographic algorithms, integration of lightweight detection and prevention algorithms, more effective detection of new attacks and sending of alarms to warn of abnormal behaviors.

4.6. Challenges and solutions

Controller Area Network was originally designed to perform communication on in-vehicle network in an easy and effective way, but not all aspects were considered. At the start, vehicles were designed as an isolated entity where communication with others was not performed. Now, new requirements in communication systems are needed. Although different research has been done to improve CAN bus security, there are several challenges that need to be addressed, including susceptibility to spoofing, fuzzy, replay, and masking attacks, compatibility of proposed solutions, metrics to measure performance.

According to authors [28], there are some security challenges for CAN bus among which are the following:

- Broadcast Nature. Since ECUs can send data frames to each other on the bus, then, a device connected to the network can spy and send on information indiscriminately.
- Fragility to DoS. A device could send high priority data packets that prevent the use of frames transmitted by other devices on the bus being processed.
- No Authenticator Fields. Since the CAN bus does not contain authentication fields, some devices could use these fields indiscriminately and send packets to another component to control it.
- Real time communication. Since it is a protocol for real-time applications, all encrypted and authenticated information should be sent with the messages without altering in real-time responses of the protocol.

Some of the security services for CAN bus are:

- Data integrity. It consists of the accuracy and validity of data, it guarantees that originally transmitted data is received.
- Authentication. It is the act of confirming the identity of a user in the system.
- Confidentiality. It means that only authorized people can access the data.
- Non-Repudiation. Guarantees that someone cannot deny the validity of the information transmitted.
- Availability. The authorized user can access the network all the time.

Many proposals related to security prevention and intrusion detection systems do not present all the possible parameters or metrics, such as accuracy, F1 score, or recovery, neither computational costs. Because of that, Table 7 shows a general analysis based on the performance for different IDS proposals using different metrics on the CAN bus, and shows if these were tested in simulation or in real environments under specific conditions. Analyzing the results based on the metrics works [101,103,110] present high values and also implementations in real environments.

It is important to mention that most IDS proposals use different data sets containing different attacks, and in the case of experiments in real environments, there are not the same constraints. Currently, it is important to evaluate how good a solution is, based on different metrics, because there are different solutions to detect and mitigate a specific set of attacks, but in many cases, the tests are not performed in real environments or these involve a short number of attacks, which does not allow to ensure that all automotive requirements are met for all cases. Therefore, it is challenging to test the methods in the same environment using automotive requirements. In addition, it is important to properly implement system responses according to the type of attack.

On the other hand, based on the previous design, Bosch 2012 officially introduced a new version of the CAN bus called CAN with a flexible data rate (CAN-FD). CAN-FD is a serial communication protocol based on the CAN bus, which supports real-time control applications with higher levels of security [112].

Table 5
Cryptography solutions for preventive security on the CAN bus.

Year	Author	Technique	Attacks	Contribution
Centralized security architecture				
2017	Z. King et al. [80]	HMAC, SHA1	DoS and replay attacks	A scheme based on message authentication using an HMAC and timestamp to prevent DoS attacks.
2015	H. Ueda et al. [84]	SHA-256, HMAC	Spoofing attacks	A centralized authentication system in CAN with improved CAN controller.
2019	Z. Lu et al. [86]	RC4, AES-128, SHA	Brute-force attack, eavesdrop attack, replay attack, masquerade attack, flooding attack	Lightweight Encryption and Authentication Protocol (LEAP) for CAN bus.
2017	A. S. Siddiqui et al. [89]	Elliptic Curve Cryptography Diffie–Hellman (ECDH), PUFs		The proposed framework is based on ECDH and it requires no shared keys to be stored on non-volatile memory within the nodes/ECUs.
2020	T.-Y. Youn et al. [90]	Session keys, MACs	Impersonation and replay attacks	A sender authentication and key management schemes considering the limitations of in-vehicle CAN.
Distributed security architecture				
2012	Lin et al. [79]	MAC, the pair-wise symmetric secret keys	Masquerade and replay attacks	A scheme based on different parameters, ID, MAC, and shared secret between transmitter and receiver.
2014	K. Han et al. [83]	A cryptographic message authentication code (MAC),	DoS attack (flooding attack, starvation attack)	An ID-Anonymization for CAN, IA-CAN protocol to protect of DoS attacks.
2015	M. Raashid et al. [91]	CRC, lightweight stream cipher	Masquerade attacks	A method to exploit a built-in fault confinement mechanism and detect masking attacks on the CAN bus.
2018	Y. Gui et al. [92]	MAC, whitelisting, blacklisting	Man in the middle, compromised device and Denial of Service (DoS) attack	A hardware-based framework with Trusted Platform Modules (TPM) enabled for secure boot, traffic control for secure communication.
2017	S. Fassak et al. [88]	Elliptic curve cryptography (ECC), establishing session keys, HMAC		A scheme based on elliptical curves, which are implemented in an adapted protocol with a series of parameters on the CAN bus.
2017	Wael A. Farag [81]	A dynamic symmetric key management	Replay attacks	CAN tool based on an algorithm that encrypts the 8-byte payload data using a symmetric key that is being dynamically changed using synchronized key generators across all nodes.
2017	P. Noureldeen et al. [82]	Light weight authentication	Denial of service attack	A lightweight CAN authentication protocol (LCAP).
2017	A. Tashiro et al. [85]	Message authentication code (MAC), key management	tampering and replay attacks	A protocol based on message authentication for CAN.
2019	M. Zhang et al. [87]	AES-128, GHASH	Replay, spoofing and sniffing attacks	A technique for assigning different priorities to the encrypted CAN frames to compensate for the increased delay.
2019	S. Woo et al. [93]	CAN ID shuffling technique, Authenticated Key Exchange Protocol 2 (AKEP2), AES-128, SHA-256, HMAC	Impersonation attack and a replay attacks	A scheme that dynamically shuffles the attack surface using the one-time ID.
2020	H. Mun et al. [94]	Hash message authentication code (64 bits HMAC), Diffie–Hellman key distribution process	Injection attacks, Impersonation Attack, Replay attack, Bus-off attack, Dos attack	A scheme based on hash message authentication code (HMAC) in specific messages to providing secure on CAN bus.
2020	B. Groza et al. [95]	Message Authentication Code (MAC), AES-128	Injection attacks, replay attacks and fuzz testing, concatenation attacks	A scheme based on an ordered CMAC buffer to authenticate the identifiers of CAN frames.
2020	Lenard, et al. [96]	MAC, Bloom filters, symmetric key cryptography, SHA-256	Man in the middle, replay attack	A MiXed data authentication for CAN bus by means of the attributes of the Bloom filters.

This new version of the CAN bus was developed taking into account two critical challenges: avoid delays in critical messages and maintain a practical length of can wires. The first one refers to how to transmit more data without delaying the processing of critical data on the bus. The second one to be able to increase transmission speed taking into account characteristics of the cable length. Considering these requirements and being able to process more information in less time, CAN-FD was designed, allowing the functionality of the CAN bus, as well as improving these limitations.

There are some changes in the CAN-FD data frame, one of which is that two control bits are agreed, one to activate the new frame format with different data length encoding and the second, to change optionally to a faster bit rate. The first cars with CAN-FD are expected to appear in 2020. Fig. 9 shows the structure of the base frame of CAN-FD protocol; different from CAN, CAN-FD can increase the bit rate, using more bytes for data.

For greater security, features of this protocol add error detection, signaling, and self-test of data, among these, are the following:

- Two new CRC polynomials to ensure error detection in longer data frames, with the same Hamming distance, as on the CAN bus.
- Bit Stuffing.
- Message Frame Check.

Table 6
Related work on other algorithmic solutions for detection attacks on the CAN bus.

Year	Author	Technique	Attacks	Contribution	TPR performance
Centralized security architecture					
2016	M. Gmiden et al. [98]	Data frame time, ID	injection attacks (random), spoofing attack	An IDS based on time arrival of data frames in CAN bus.	-- -- --
2018	A. J. Brown et al. [99]	Identifiers of CAN bus	Denial of service (DoS) attack	A method to identify a malicious node and denied access to the bus after a sequence of consecutive messages. (D/P)	-- -- --
2018	Wang et al. [100]	Entropy	Injection attacks, flooding attack, Weak Injection	A IDS based on the entropy of the identifier bits in CAN messages.	100–91%
2019	NING et al. [101]	LOF (Local Outlier Factor)	Bus-off attack, spoofing attack, wireless attack	Intrusion detection method based on Local Outlier Factor to detect attacks in ECUs.	99–90%
2020	Ohira et al. [102]	Divider circuit for sampling delay time	Spoofing attacks	A delay-time based sender identification method of ECUs.	96.7%
2020	S. Katragadda et al. [103]	Sequence mining approach, pattern-frequency recognition	Replay attacks	A sequence mining approach to detect low-rate injection attacks in CAN bus.	100–98%
2020	Tariq et al. [104]	A rule-based parametric approach crafted from analyzing dynamic network traffic characteristics	Hazardous DoS, fuzzing, and replay attacks	A comprehensive anomaly generation, detection, and evaluation system (CAN-ADF).	99%
2020	Murvay, et al. [105]	Signal arrival times	spoofing and replay attacks	An IDS based on monitoring the propagation time of the physical signals sent on the bus.	100%
Semi-distributed security architecture					
2019	B. Groza et al. [97]	Bloom filters, frame periodicity	Attack free state, injection attack, replay and modification attacks	IDS using Bloom filtering to test frame periodicity based on message identifiers and parts of the data-field in CAN bus.	-- -- --
Distributed security architecture					
2012	T. Matsumoto et al. [106]	Flags and error frames	unauthorized packets	A prevention method for unauthorized data transmission in CAN (D/P)	100%
2018	Kwon et al. [107]	Mitigation manager, reconfigurable ECU, data frame bits	DoS attacks, unauthorized packets, or unauthorized control commands	A method based on a mitigation manager and a reconfigurable mechanism to mitigate attacks.	-- -- --
2019	Tian et al. [108]	temperature varied clock offset fingerprint.	masquerade attacks	The concept of variable temperature ECU fingerprints is proposed to improve the accuracy of source identification in real-world vehicle CAN intrusion cases.	96%
2020	Cheng et al. [109]	Moving Target Defense (MTD), shifting of frame IDs and white list	Spoofing attacks, injection attacks	A decentralized mechanism guaranteeing that the protection could be done simultaneously without additional communication.	100%

Table 7
Attack detection analysis for different proposals based on algorithmic performance metrics.

Proposal	Accuracy	F1-score	Recall	Precision	Simulation	Real environment
A. Alshammari et al. [73]	98–96	93.5	99–96	100–96	✓	
H.M. Song et al [68]	-- -- --	99.9	99.8	99.8	✓	
S. Katragadda et al. [103]	100–98.68	99.7–99.09	100–85.12	100		✓
Barletta et al. [71]	100–99.75	100–96.70	100	100–94.42	✓	
Tariq et al. [104]	99.45	100–95	100–93	100–97	✓	
Wang et al. [75]	-- -- --	-- -- --	81	99–98	✓	
Ning et al. [101]	97	-- -- --	-- -- --	-- -- --		✓
J. Xiao et al. [69]	-- -- --	96.0	96.0	97.0	✓	
J. Zhou et al. [110]	99	99.72	99.69	99.76–98.37		✓
Olufowobi et al. [111]	98	98–87	100–81	97–80	✓	

- Using detection mechanism, different types of errors can be detected (local errors, global errors and up to 5 random errors in a frame).

Some works have been carried out based on methods to improve security over CAN-FD. For example, in [89], the authors proposed a framework based on lightweight hardware based on authentication and secure key exchange over CAN-FD bus. The approach consists of hardware based on security enhancements to each connected ECU. According to the experiments, the method improves security at a device level and communication allowing detection of any modifications or invasive attacks to legitimate ECUs.

Since the bandwidth used in the CAN bus is limited, devices with low computer processing, such as microcontrollers, required other communication protocols in order to allow users to access other services and operation of real-time multimedia applications. Protocols such as

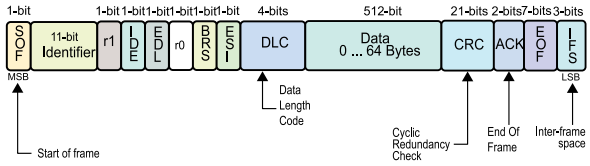

Fig. 9. CAN-FD base format frame structure.

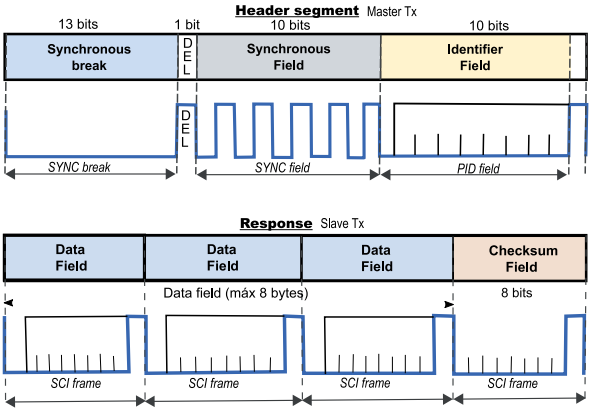

Fig. 10. LIN frame structure.

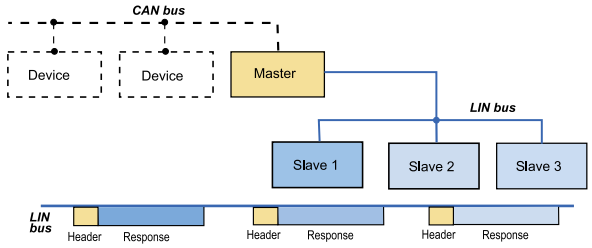

Fig. 11. LIN Network structure.

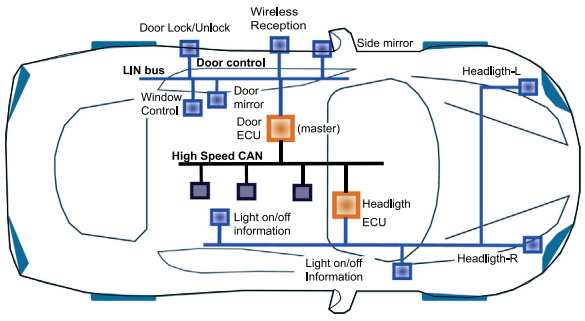

Fig. 12. LIN network connected to CAN bus within a car.

FlexRay have been created to improve the performance of networks in entertainment applications. This protocol can be used as a data backbone with other protocols, such as CAN, MOST and LIN, and allow more versatile functionality in cars; although this implies more vulnerable points in the network that can be used by hackers to access information and provoke failures in-car systems.

5. Security solutions on other network protocols

A brief description of other protocols used in the automotive industry will be presented as well as its security issues and proposed solutions in the literature.

5.1. LIN bus

Local Interconnected Network (LIN) was proposed in 1998 by a consortium of automotive companies (BMW, Audi, Daimler Chrysler, Mercedes-Benz, Volcano Automotive, and Volkswagen) together with Motorola. The aim was to create a standard and a low-cost bus. With its implementation, some applications emerged in car seats, door locks, mirrors, etc.

LIN bus consists of a sub-bus system based on a serial communications protocol, which is able to perform single master/multiple slave functions. In addition, time synchronization is used to send or receive messages from devices. An error detection mechanism based on data checksum and parity check is also used [113].

Each LIN frame consists of two main parts: (1) A header is always transmitted by the master device. The token is divided in three fields: sync break, 1 delimiter bit, sync field, and a protected identifier (PID), and (2) data block (response) is the frame sent by the slave device. The answer consists of up to 8 bytes of data and a checksum. Fig. 10 shows detailed structure of LIN bus data frame.

Fig. 11 shows a typically LIN network, which is composed of a master that communicates with other slave devices. This scheme leads to a master communicating with a set of devices based on time triggering events and, at the same time, this master can be part of another network such as CAN bus. LIN supports from 2 to 16 nodes for data communication.

Some characteristics of LIN bus are:

- Low-cost communication system.
- Widely used in cars production.
- Typically found in body and comfort systems.
- Error detection: Checksum-Parity bits.
- A communication cable at 20 Kbits/sec, up to 8 bytes of payload.
- 64 different identifiers available, which denote content msec.
- It can control different functions: seating position motors, occupancy control, roof, steering wheel, mirrors, windshield wipers, climate controls, etc.
- It is based on a time-triggered scheduling with guaranteed latency time.

Fig. 12 shows the structure of a network that is composed of CAN and LIN buses, in this case, the union between these buses is done through an ECU, which allows greater functionality in the car at different transmission speeds depending on required services.

5.1.1. LIN bus security issues and solutions

According to characteristics of the LIN protocol, some security issues can be mentioned: low-security detection mechanism, unencrypted messages, limited architecture, dependency on slaves and single master, broadcast transmission, and restriction to non-critical functions.

There is little work focusing on solving security problems on the LIN bus. In [114], authors present a LIN bus security analysis and considerations. For example, disadvantages of the LIN bus versus CAN in networks and how security can be improved. Experiments show how an unauthorized device can destroy any message packet during master/slave communication, it also shows that it can be used to create erroneous but valid packets. This scenario is used by [115] to stop communication, they experimentally performed some attacks in a vehicle microcontroller; and present countermeasures against them. For the response collision attacks, they proposed a Byte Assignment in Response, setting the significant data to the first byte of the response; and for the header collision attack, a message authentication code is proposed. Additionally, they suggest sending an abnormal signal when an error is detected.

Authors in [116] point out that by spoofing messages LIN bus can be compromised, they present two scenarios; the first one by only attacking the LIN master malicious, sleep frames can deactivate the subnet;

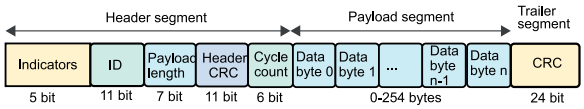

Fig. 13. FlexRay frame structure.

and the second one by sending frames with bogus synchronization bytes will disable the LIN Network.

It is important to examine the implementation of security mechanisms on the LIN bus, even though its applications on vehicle communication are not considered crucial. An attack on this bus can produce malfunction in the vehicle or even avoid the driver's entrance to it.

Next, a description of FlexRay and an analysis of security issues and solutions are presented.

5.2. FlexRay bus

In automobiles, several devices are responsible for performing electronic monitoring and control (ECU). Since the number of ECUs in automotive systems grows with the increasing functionality in cars. New services such as telematics and automotive assistance like ADAS have been added, thus real-time communication networks have become mandatory. FlexRay was designed to target those challenges.

FlexRay was proposed in 2000 and was developed by Daimler Chrysler, BMW, Freescale, and Phillips. FlexRay is defined as a high bandwidth communication protocol intended for high-speed technology in automotive vehicles. This protocol supports single or dual-channel configurations, which involves one or two pairs of wires, respectively. A differential signaling is used to reduce external noise effect on the network, [117].

In Fig. 13 FlexRay frame structure is detailed. FlexRay frame has three main elements: header, payload, and forward. The first 5 header bits set basic properties in the data frame. Frame ID defines slot position in a static segment (critical information) and indicates frame priority for a dynamic segment (low priority data). Payload length is defined as data length. Cyclic redundancy check (CRC) header is calculated with a synchronization frame indicator, with a start frame indicator, frame identification and useful load length [118]. Cycle count is a serial number of the locally defined frame at a node. Trailer segment sets CRC, which is calculated on payload and header segments [119].

Among main features in FlexRay protocol are:

- Time triggering protocol.
- Deterministic behavior.
- Fault tolerance.
- Bus, star, multi/sta network configuration expandable up to 64 nodes.
- Hard real-time systems usage in applications such as chassis, brakes direction, driver assistance.
- Cyclic data transmission control based on Time Division Multiple Access (TDMA) and flexible TDMA (FTDMA).

FlexRay is used for data backbone composed of different buses such as CAN, LIN, MOST, etc. It is implemented in critical applications and integrated into communication with other protocols. Commonly, there are three levels in FlexRay systems: network, interface, and protocol engine. At network-level, single/dual channel and different topologies: bus-type, star type, and hybrid type are defined. At the interface level a guardian bus and physical interface, together with error detection and containment in the time domain are determined. At a protocol engine level, the control host interface (CHI) and protocol engine are setups.

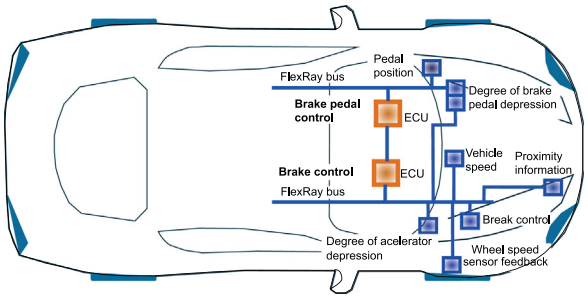

Fig. 14. Brake pedal control and brake control systems using a FlexRay network.

5.2.1. FlexRay security issues and solutions

Following the previously discussed taxonomy for CAN-bus that relates architectural type (centralized, semi-distributed, and distributed) and computational algorithmic techniques to tackle security issues and to provide solutions for FlexRay protocol are herein discussed. It is worth mentioning that similar to CAN bus, the main algorithmic researched techniques are Neural Networks (NN) and cryptographic techniques as well as a handful of other implementation approaches using a variety of processing platforms and design models.

In [120,121], a FlexRay network monitoring control based on Neural Networks (NN) for security and reliability is proposed. NN are used as model predictors to dynamically adapt heavy loads affecting stability and performance.

In [122], authors analyze FlexRay scheduling and characteristics such as static and dynamic segments and implementation verification.

In [119] authors considered vulnerabilities and a hidden attack on FlexRay. An Advanced Encryption Standard (AES)-128 and SHA-1 algorithms are implemented. In their proposed protocol they use AES-128 and HMAC to achieve confidentiality and authentication, respectively. They used CANoe software system from Vector company to simulate a FlexRay network with two virtual ECUs. Empirical validation demonstrates that their protocol is suitable for FlexRay networks.

In [123], authors adjust a CAN bus protocol adding a lightweight authentication mechanism to FlexRay and achieving encrypted and authenticated messages transmission. FlexRay provides more flexibility in comparison to CAN bus while maintaining robustness.

In [124], a transmissions adaptive authentication protocol is proposed by taking advantage of time-triggered communication in FlexRay. The authors also considered non-deterministic transmissions due to FlexRay dynamic segment.

A distributed FlexRay based approach to deploy computationally expensive authentication algorithms is proposed in [125]. An optimization problem to determine the minimal number of co-processing units required to fulfill security and deadlines is targeted. FPGAs and ASICs processing platforms as co-processing units.

In [126–128], new FlexRay improved architectures have been proposed. In [128], authors proposed a CAN/FlexRay gateway using hardware/software (HW/SW) co-design. Their proposed architecture has three main modules: receive, convert, and send modules for communication between CAN and FlexRay interfaces. Experimental results show their system can reduce in 94.7% the execution time for in-vehicle networks.

Other works, such as [129] are focused on communication controllers for FlexRay protocol, designed and implemented on FPGAs. The proposed controller is suitable to perform communication in FlexRay networks and has configurable feature extensions to provide functionality that is unavailable in standard implementations or off-the-shelf devices. In Fig. 14, brake pedal and brake control systems connected through a FlexRay network are drawn.

Strong determinism and high fault tolerance for critical applications are significant characteristics in FlexRay protocol. Pullen et al. included an authentication mechanism in the optional dual-channel

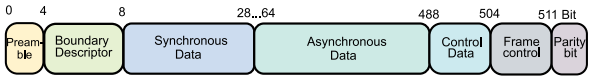

Fig. 15. MOST25 data frame structure.

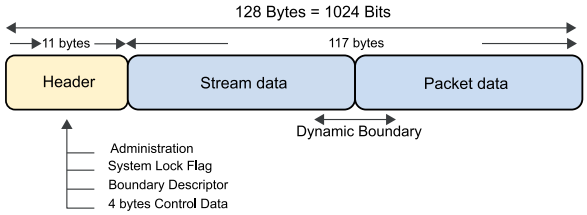

Fig. 16. MOST50 data frame structure.

mode for backward operation and also proposed several techniques for cryptographic keys managing and authentication [130]. In [131], defense techniques based on authentication tags splitting using independent channels against spoofing attacks are developed, the concurrent transmission allows for timing and overhead reduction.

A common perception among FlexRay research works is that it covers a number of characteristics lacking in CAN bus, thus the automotive industry is steering towards adopting FlexRay either in a standalone fashion or in combination with CAN bus and other protocols to increase security.

5.3. Most oriented system transport bus

Most Oriented System Transport (MOST) bus was initially proposed and discussed at BMW, in collaboration with Daimler Benz, Becker (Harman Automotive Division), and OASIS Silicon Systems (SMSC) on the basis of the D2B system, developed around 1996 [132]. The initial phase sought specification and collaboration in network function. MOST collaboration focuses on developing specifications and promoting technology. Different companies such as Audi, BMW, Mercedes, and Volkswagen incorporated the MOST technology in their cars.

Main applications of MOST protocol are oriented to multimedia applications. Some characteristics of MOST bus are as follows:

- MOST peer-to-peer network is connected by a plug and play with up to 64 nodes through a star, bus, or ring topology.
- Control channel can request or release most devices on one of the sixty configurable data channels.
- Type of error detection: CRC.
- Its operation is based on a fiber-optic bus.
- Applications: multimedia, entertainment, navigation, services information, DVD, MP3, GPS systems, etc.
- High throughput and dynamic bandwidth: MOST has synchronous /asynchronous channels that can allocate a part of the bandwidth for required services.

There are three different versions of MOST protocol:

MOST25 a 64 bytes frame, MOST50 a 128 bytes frame and MOST150 a 384 bytes frame. Fig. 15 shows the structure of MOST25 data frame. The beginning (preamble) and the end (parity bit) of MOST25 frame contain the control information of the data frame, while 16 frames are combined in one block to transport the control message.

MOST50 achieves a bit rate of 50 Mbit/sec. The frame length increases to 128 bytes but the sample rate remains the same. MOST50 frame contains a header composed by administration, system lock flag, boundary descriptor, and control data, a target address, source address, CRC, stream data and packet data. Fig. 16 shows MOST50 protocol frame details.

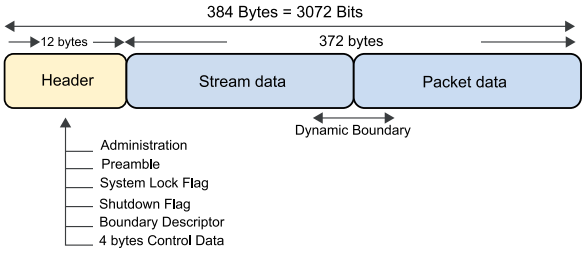

Fig. 17. MOST150 data frame structure.

MOST150 data frame is drawn in Fig. 17. MOST150 uses the same sample rate and three times higher baud rate as MOST50. MOST150 data frame starts with a header (control data and boundary descriptor), thus 372 bytes are used to stream data and packet data (target address, source address, CRC).

Various research study and propose MOST implementations in-vehicle networks [133–136], while other works have focused on proposing the implementation of devices that use MOST together with other protocols such as CAN and FlexRay [136–138].

5.3.1. MOST bus security issues and solutions

Differently from other automotive buses, the sender and receiver address is always included in a MOST message. An internal system service detects errors over parity bits, status flags, and checksums; determining if an erroneous node should be switched off [115].

In a MOST network, one MOST device sends timing frames to synchronize MOST slaves; in an attack malicious timing frames can be sent to disrupt synchronization. The MOST protocol allows for bandwidth contention. Communication between MOST devices depends on message priority in a fixed-length communication segment (Dynamic Segment). A malicious MOST device can jam the segment by spoofing high-priority messages [116].

Authors in [113] point out that some security elementary practices could be included to guarantee secure vehicular bus communication, including MOST, like controller authentication, encrypted communication, and including gateway firewall. Authors in [13] suggest that protection against the synchronization disruption and jamming attacks should be explored.

As MOST is a bus linked to the infotainment in the vehicle, implementing security on it should be taken into consideration. Many attacks on vehicles have been through the infotainment system, and have exposed the data and safety of the driver.

5.4. Ethernet

Nowadays, cars communicate with each other and other devices through a wired or wireless network. There are different ways to implement vehicular networks, among these are: Networking into the car to use system entertainment, control systems, etc. Communication between cars avoids collisions, allows networking between cars and the road for greater safety and car efficiency, and finally, connection to the internet allows passengers to use maps for navigation and traffic updates.

Automotive Ethernet (AE) was launched by BMW's X5 around 2013, more than 40 years after the Ethernet was proposed. In Fig. 18, the Ethernet data frame is shown, it is composed by a preamble, a destination address, a source address, type of length, 46–1500 bytes for payload, and CRC. Addresses are locally unique, this means the same address cannot be employed in the LAN.

Today the automotive industry Ethernet is based on Audio Video Bridging (AVB). AVB is a method for transporting audio and video transmissions over Ethernet-based networks. AVB is based on a set of IEEE standards for Ethernet networks that defines transport, signaling, and synchronization of transmitted video and audio.

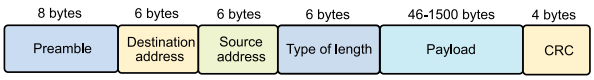

Fig. 18. Ethernet protocol frame structure.

An important aspect is the quality of service (QoS) because it allows specific services such as voice and video traffic to have priority over data traffic to ensure delivery time. Similarly, AVB adds flow signaling, automatic bandwidth reservation, synchronization, and traffic prioritization.

AVB Ethernet has some advantages over other communication protocols, among them, are:

- Ethernet is highly flexible and scalable.
- It is compatible with all types of communication flows.
- Cost of Ethernet switches per port and bandwidth remains low.
- Ethernet is a standard used anywhere, so companies can take advantage of existing networks and cabling.
- It follows a plug and play fashion for easy network implementation.
- AVB is based on a set of IEEE standards, which guarantees interoperability between different vendors.

Automotive Ethernet (AE) offers a superior bandwidth to other automotive protocols, thus applications requiring more resources can provide better passengers experiences. Fast technological development allows highly demanding protocols therefore Ethernet may be the future in automotive communications.

There are several studies and implementations for Ethernet in Automotive communications [139,140], including some focused on communication between Ethernet and other protocols like CAN and FlexRay [141].

Ethernet is efficient in terms of bandwidth and interoperability [142], providing faster transfer rates and larger payloads than other automotive protocols, hence, the physical layer standard BroadR-Reach has emerged as the future of automotive applications. Through OPEN Alliance, the automotive industry has standardized the BroadR-Reach point-to-point physical layer (PHY) as the standard for Ethernet communication in vehicles [143]. BroadR-Reach uses bidirectional communications similar to 1000BASE-T Ethernet, where both interfaces communicate simultaneously over the same twisted pair cable.

Some benefits of BroadR-Reach are the reduction of up to 30% of the wiring weight of the vehicle and a lot of technology can be quickly transferred from the telecommunications industry [142]. The Ethernet protocol is also evolving to guarantee latency requirements of the vehicle industry.

SOME/IP (Scalable service-Oriented MiddlewarE over IP) is an AUTOSAR automotive/embedded protocol that supports remote procedure calls, event notifications, and the underlying serialization/wire format [144]. SOME/IP is the first protocol developed for automotive use, utilizes unicast and multicast communication [145]. SOME/IP establishes three ways of communication between server and client [57]: (1) **Events**, in this type, clients subscribe to certain required information from a provider and this one transmits the information to the clients at a specified time interval or if a value has changed. Mainly used for information that is only valid for a short time; (2) **Fields**, allows the field's definition to be read or modified by clients using *get* and *set* methods. Fields are meant for status-like properties that refer to history; (3) **Methods**, allows a client to execute a method provided by the server. Some advantages of SOME/IP are its compatibility with many communication partners, it is compatible with AUTOSAR on the wire-format level, is scalable from tiny to large platforms [144]. A disadvantage of SOME/IP is the lack of security functionalities, which leaves the transmitted data completely vulnerable to malicious attacks.

5.4.1. Ethernet security issues and solutions

One of the main problems of security on Ethernet networks consists of the compromise of administrators because each new software installation, switches, and configuration requires to meet the security policy. Due to these possible deficiencies, an attacker could test the defenses of the Ethernet network freely until vulnerable points are found. There are different reasons why attackers may utilize network access for: learning network topology, know the traffic for later attacks, taking control over switches, routers, manipulation of information, etc.

Some works are focused on the security of ethernet networks. For example in [146], authors explain the main attacks over Ethernet LAN. According to the authors, intrusion detection and prevention systems can detect several attacks: DHCP snooping pairs, MAC addresses to IP addresses, VLAN double tagging, thwarts ARP spoofing, etc. In [147], authors propose new network architecture for in-car audio/video communication based on the Ethernet technology.

Other authors [148] propose an IDS to detect attacks on audio–video transport protocol (AVTP) in Ethernet networks. This method is based on Convolutional Neural Networks (CNN). To test their method authors built a physical BroadR-Reach-based and captured real AVTP packets. Their results show a F1-score and recall over 0.97 and 0.99, respectively.

Because of its early application on in-vehicle communication, the implementation of security schemes is viable. New applications should consider security integration from the beginning.

6. Future research directions

Currently, there is ongoing research related to security solutions for in-vehicle protocols, however, implementation of most of these security methods needs to be reported considering real scenarios in order to guarantee an efficient performance in vehicles. Based on the current horizon of in-vehicle protocols, the main future research directions are as follows:

- Algorithms used on in-vehicle solutions. There are different algorithms used for security solutions based on neural networks, cryptography, machine learning, etc. Although there are many proposed solutions, it is necessary to face new challenges, among which are:
  - To propose solutions for detecting and preventing a large number of attacks and consequences in car control.
  - To combine research areas to improve the security performance.
  - To develop lightweight algorithms to improve the performance and reduce the computational requirements under real-world conditions.
  - To reduce the busload used for the proposed methods.
  - To implement robust key generation algorithms for methods based on cryptographic techniques.
  - To create databases of in-vehicle systems in real conditions to carry out tests with the algorithms proposed in a wider cyber attack spectrum.

- Intrusion detection system. IDS's based on different schemes have been proposed with good results, but only few works are implemented in real conditions on vehicle networks and countermeasures are presented. Based on current research, some directions are as follows:
  - To increase solutions performance for a large set of cyber attacks based on different types of metrics (accuracy, recall, precision, F-score, etc.).
  - To analyze size impact for training data sets on the detection performance and computation resources of the algorithms.
  - To improve false-positive percentages of detection attacks in proposed IDS.

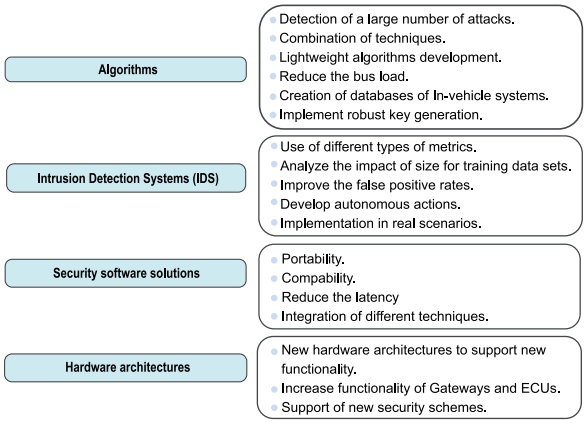

Fig. 19. Future research directions for security on in-vehicle protocols.

  - To develop autonomous actions of the systems to indicate and to mitigate attacks presence.

- Security software solutions. The solutions proposed to improve security have shown good results, but also the new challenges demand that these solutions comply with the new requirements and paradigms used.
  - Test the proposed methods in real environments to achieve portability and compatibility.
  - Cyber security standards. In recent years, there were no specific cyber security standards for vehicles because, at present, new standards will impact the development and safety of smart vehicles.
  - Reduce the waiting time to methods in real-time applications.

- Security hardware architectures. Architectures used on in-vehicle networks were originally designed only to support basic schemas, however, robust architectures are now required to support new security schemas.
  - New hardware architectures to support new functionality are also needed.
  - Increase functionality of gateways and ECUs to prevent the transmission of suspicious messages on the network.
  - Hardware architectures to support autonomous functions and algorithms.

Fig. 19 shows some relevant future directions for four areas in security of in-vehicle protocols.

In the future, given the autonomy required in cars and their intercommunication with the environment, safety will continue to be a fundamental aspect to ensure its proper functioning.

7. Conclusions

The constant progress of technology has impacted current vehicles which have more and more sensors, electro-mechanical devices, and electronic control units, which allow them to have more functionality and provide drivers with more services.

In this work, the main characteristics of in-vehicle protocols were reviewed, as well as the main recent works related to security technologies, security architectures, and cyberattacks generated to the main automotive communication protocols. We classified the proposed works based on the type of security architecture, true positive rate, and technique used. Although at the beginning protocols such as the CAN bus was not designed to withstand attacks, current requirements demand new schemes that provide greater security. These schemes propose different solutions, which range from adding cryptographic algorithms to adding identifiers in data frames to improve security in the original versions of the communication protocols.

In the case of the CAN bus, several works have been published where security schemes to detect software-based attacks using cryptography, neural networks, machine learning, and other algorithms are proposed. In most of the proposed works, the use of identifiers, detection of alterations in the sending time, authentication of messages, among others are presented. Other schemes are based on hardware architectures to detect modification of the transmission clock speed, analysis of deviations of the entropy information, changes in the speed of transmission of messages, etc.

On the other hand, most speed applications are implemented in cars, therefore, more complex devices are interconnected through automotive networks. Also, communication protocols require security mechanisms to improve security information and increase speed in data transmission. Solutions based on FlexRay, LIN, and Ethernet with CAN bus have been proposed, and the most robust schemes can help to improve functionality in cars. Some of the current challenges of communications protocols are bandwidth, latency, cost, attack detection and countermeasures, compatibility, and security. This survey involves the most recent works related to the proposed security methods, classification as well as the criteria to be considered in the improvement of the security on in-vehicle communication protocols.

CRediT authorship contribution statement

Alfonso Martínez-Cruz: Conceptualization, Methodology, Investigation, Writing - original draft. **Kelsey A. Ramírez-Gutiérrez:** Investigation, Visualization, Writing - original draft. **Claudia Feregrino-Uribe:** Resources, Writing - original draft, Data curation, Writing - original draft. **Alicia Morales-Reyes:** Resources, Writing - original draft.

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.

Acknowledgment

This work has been partially supported by grant under agreements CONACyT, Mexico: 882 “Desarrollo de Sistemas de Seguridad para Aplicación en la Industria Automotriz en el Edo de Tlaxcala”.

References

[1] J. D’Ambrosio, G. Soremekun, Systems engineering challenges and MBSE opportunities for automotive system design, in: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2017, pp. 2075–2080.

[2] M. Scalas, G. Giacinto, Automotive cybersecurity: Foundations for next-generation vehicles, in: 2019 2nd International Conference on New Trends in Computing Sciences (ICTCS), 2019, pp. 1–6, http://dx.doi.org/10.1109/ICTCS.2019.8923077.

[3] N. Nicolas, F. Simonot-Lion, In-vehicle communication networks - a historical perspective and review, in: Industrial Communication Technology Handbook, University of Luxembourg, vol. 1, 2013, pp. 1–50, http://hdl.handle.net/10993/5540.

[4] URL: https://www.freepik.es/vector-gratis/coche-deportivo-rojo-aislado-vector-blanco_3529810.htm#page=1&query=car&position=39.

[5] N. Nowdehi, A. Lautenbach, T. Olovsson, In-vehicle CAN message authentication: An evaluation based on industrial criteria, in: 2017 IEEE 86th Vehicular Technology Conference (VTC-Fall), 2017, pp. 1–7, http://dx.doi.org/10.1109/VTCFall.2017.8288327.

[6] 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), 2018, pp. 201–205, http://dx.doi.org/10.1109/iCCECOME.2018.8658720.

[7] B. Groza, P. Murvay, Security solutions for the controller area network: Bringing authentication to in-vehicle networks, IEEE Veh. Technol. Mag. 13 (1) (2018) 40–47, http://dx.doi.org/10.1109/MVT.2017.2736344.

[8] V.H. Le, J. den Hartog, N. Zannone, Security and privacy for innovative automotive applications: A survey, Comput. Commun. 132 (2018) 17–41, http://dx.doi.org/10.1016/j.comcom.2018.09.010, URL: http://www.sciencedirect.com/science/article/pii/S014036641731174X.

[9] C. Young, J. Zambreno, H. Olufowobi, G. Bloom, Survey of Automotive Controller Area network intrusion detection systems, IEEE Design Test (2019) 1, http://dx.doi.org/10.1109/MDAT.2019.2899062.

[10] S. Lokman, A.T. Othman, Abu-Bakar, Intrusion detection system for automotive controller area network (CAN) bus system: a review, Wirel. Com Netw. (2019) 1 (1) (2019) 1–17, http://dx.doi.org/10.1186/s13638-019-1484-3.

[11] M. Gmiden, M.H. Gmiden, H. Trabelsi, Cryptographic and intrusion detection system for automotive CAN bus: Survey and contributions, in: 2019 16th International Multi-Conference on Systems, Signals Devices (SSD), 2019, pp. 158–163.

[12] W. Wu, R. Li, G. Xie, J. An, Y. Bai, J. Zhou, K. Li, A survey of intrusion detection for in-vehicle networks, IEEE Trans. Intell. Transp. Syst. (2019) 1–15, http://dx.doi.org/10.1109/TITS.2019.2908074.

[13] Z. El-Rewini, K. Sadatsharan, D.F. Selvaraj, S.J. Plathottam, P. Ranganathan, Cybersecurity challenges in vehicular communications, Veh. Commun. 23 (2020) 100214, http://dx.doi.org/10.1016/j.vehcom.2019.100214, URL: http://www.sciencedirect.com/science/article/pii/S221420961930261X.

[14] P. Urien, Designing attacks against Automotive Control Area network bus and electronic control units, in: 2019 16th IEEE Annual Consumer Communications Networking Conference (CCNC), 2019, pp. 1–4, http://dx.doi.org/10.1109/CCNC.2019.8651708.

[15] K. Iehira, H. Inoue, K. Ishida, Spoofing attack using bus-off attacks against a specific ECU of the can bus, in: 2018 15th IEEE Annual Consumer Communications Networking Conference (CCNC), 2018, pp. 1–4, http://dx.doi.org/10.1109/CCNC.2018.8319180.

[16] X. Zhang, W. Feng, J. Wang, Z. Wang, Defensing the malicious attacks of vehicular network in runtime verification perspective, in: 2016 IEEE International Conference on Electronic Information and Communication Technology (ICEICT), 2016, pp. 126–133, http://dx.doi.org/10.1109/ICEICT.2016.7879666.

[17] J.D. Florian Sommer, R. Kriesten, Survey and classification of automotive security attacks, MDPI Open Access J. 10 (4) (2019) 1–29, http://dx.doi.org/10.3390/info10040148.

[18] C. Miller, C. Valasek, Remote exploitation of an unaltered passenger vehicle, in: Blackhat 2015, 2015.

[19] A. Greenberg, Hackers Remotely Kill a Jeep on the Highway-With Me in It. [Online], 2015, Available: https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/.

[20] Upstream Security Ltd, Upstream Security: 2021 Global Automotive Cyber security Report, Technical Report, 2020.

[21] Tencent Security Keen Lab, Mercedes-Benz MBUX Security Research Report, Technical Report.

[22] P. Bigelo, Nissan disables leaf app due to hacking concerns, 2016, URL: https://www.autoblog.com/2016/02/25/nissanconnect-ev-leaf-app-hacking-followup/?guccounter=1.

[23] M. Mimoso, Hyundai patches leaky blue link mobile app, 2017, URL: https://threatpost.com/hyundai-patches-leaky-blue-link-mobile-app/125182/.

[24] S. Abbott-McCune, L.A. Shay, Intrusion prevention system of automotive network CAN bus, in: 2016 IEEE International Carnahan Conference on Security Technology (ICCST), 2016, pp. 1–8, http://dx.doi.org/10.1109/CCST.2016.7815711.

[25] J. Patel, M.L. Das, S. Nandi, On the security of remote key less entry for vehicles, in: 2018 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), 2018, pp. 1–6, http://dx.doi.org/10.1109/ANTS.2018.8710105.

[26] D.H.L. Wonsuk Choi, Sound-proximity: 2-factor authentication against relay attack on passive keyless entry and start system, J. Adv. Transp. 2018 (2018) 1–13, http://dx.doi.org/10.1155/2018/1935974.

[27] K. Greene, D. Rodgers, H. Dykhuizen, Q. Niyaz, K. Al Shamaileh, V. Devabhaktuni, A defense mechanism against replay attack in remote keyless entry systems using timestamping and XOR logic, IEEE Consumer Electron. Mag. 10 (1) (2021) 101–108, http://dx.doi.org/10.1109/MCE.2020.3012425.

[28] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, Experimental security analysis of a modern automobile, in: 2010 IEEE Symposium on Security and Privacy, 2010, pp. 447–462, http://dx.doi.org/10.1109/SP.2010.34.

[29] A. Yadav, G. Bose, R. Bhange, K. Kapoor, N. Iyengar, R.D. Caytiles, Security, vulnerability and protection of vehicular on-board diagnostics, Int. J. Secur. Appl. 10 (2016) 405–422.

[30] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, T. Kohno, Comprehensive experimental analyses of automotive attack surfaces, in: Proceedings of the 20th USENIX Conference on Security, SEC'11, USENIX Association, USA, 2011, p. 6.

[31] M. Cheah, S.A. Shaikh, O. Haas, A. Ruddle, Towards a systematic security evaluation of the automotive bluetooth interface, Veh. Commun. 9 (2017) 8–18, http://dx.doi.org/10.1016/j.vehcom.2017.02.008, URL: https://www.sciencedirect.com/science/article/pii/S2214209616301474.

[32] P. Doherty, A. Molloy, M. Glavin, F. Morgan, A review of bluetooth security in the automotive environment, 2004.

[33] E.F.M. Josephlal, S. Adepu, Vulnerability analysis of an automotive infotainment system's wifi capability, in: 2019 IEEE 19th International Symposium on High Assurance Systems Engineering (HASE), 2019, pp. 241–246, http://dx.doi.org/10.1109/HASE.2019.00044.

[34] S. Woo, H.J. Jo, D.H. Lee, A practical wireless attack on the connected car and security protocol for in-vehicle CAN, IEEE Trans. Intell. Transp. Syst. 16 (2) (2015) 993–1006, http://dx.doi.org/10.1109/TITS.2014.2351612.

[35] R. Changalvala, H. Malik, Lidar data integrity verification for autonomous vehicle, IEEE Access 7 (2019) 138018–138031, http://dx.doi.org/10.1109/ACCESS.2019.2943207.

[36] I.-P. Hwang, C.-H. Lee, Mutual interferences of a true-random LiDAR with other LiDAR signals, IEEE Access 8 (2020) 124123–124133, http://dx.doi.org/10.1109/ACCESS.2020.3004891.

[37] J. Petit, B. Stottelaar, M. Feiri, Remote attacks on automated vehicles sensors : Experiments on camera and LiDAR, 2015.

[38] P. Kapoor, A. Vora, K.-D. Kang, Detecting and mitigating spoofing attack against an automotive radar, in: 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall), 2018, pp. 1–6, http://dx.doi.org/10.1109/VTCFall.2018.8690734.

[39] R. Komissarov, A. Wool, Spoofing attacks against vehicular FMCW radar, 2021, URL: arXiv:2104.13318.

[40] R.G. Dutta, X. Guo, T. Zhang, K. Kwiat, C. Kamhoua, L. Njilla, Y. Jin, Estimation of safe sensor measurements of autonomous system under attack, in: 2017 54th ACM/EDAC/IEEE Design Automation Conference (DAC), 2017, pp. 1–6, http://dx.doi.org/10.1145/3061639.3062241.

[41] A. Francillon, B. Danev, S. Capkun, Relay attacks on passive keyless entry and start systems in modern cars, IACR Cryptol. EPrint Arch. 2010 (2011) 332.

[42] A.I. Alrabady, S.M. Mahmud, Analysis of attacks against the security of keyless-entry systems for vehicles and suggestions for improved designs, IEEE Trans. Veh. Technol. 54 (1) (2005) 41–50, http://dx.doi.org/10.1109/TVT.2004.838829.

[43] S. van de Beek, F. Leferink, Vulnerability of remote keyless-entry systems against pulsed electromagnetic interference and possible improvements, IEEE Trans. Electromag. Compat. 58 (4) (2016) 1259–1265, http://dx.doi.org/10.1109/TEMC.2016.2570303.

[44] K. Reif, Gasoline engine management, systems and components, bosch professional automotive information, Springer Fachmedien Wiesbaden 1 (1) (2015) 1–363, http://dx.doi.org/10.1007/978-3-658-03964-6.

[45] A. Maruaisap, P. Kumhom, A hardware-based security scheme for in-vehicle CAN, in: 2016 International Computer Science and Engineering Conference (ICSEC), 2016, pp. 1–5, http://dx.doi.org/10.1109/ICSEC.2016.7859891.

[46] B. Poudel, A. Munir, Design and evaluation of a reconfigurable ECU architecture for secure and dependable automotive CPS, IEEE Trans. Dependable Secure Comput. (2018) 1, http://dx.doi.org/10.1109/TDSC.2018.2883057.

[47] M. Wolf, T. Gendrullis, Design, implementation, and evaluation of a vehicular hardware security module, in: Information Security and Cryptology - ICISC 2011. ICISC 2011, in: Lecture Notes in Computer Science, vol. 7259, Springer, Berlin, Heidelberg, 2012, pp. 302–318, http://dx.doi.org/10.1007/978-3-642-31912-9_20.

[48] NXP, Automotive Gateway: A Key Component to Securing the Connected Car. Technical Report.

[49] T. van Roermund, A. Bening, F. Poulard, Cybersecurity for ECUs: Attacks and Countermeasures, Technical Report.

[50] S. Seifert, R. Obermaisser, Secure automotive gateway — Secure communication for future cars, in: 2014 12th IEEE International Conference on Industrial Informatics (INDIN), 2014, pp. 213–220, http://dx.doi.org/10.1109/INDIN.2014.6945510.

[51] J. Berg, J. Pommer, C. Jin, F. Malmin, J. Kristensson, Secure Gateway – A concept for an in-vehicle IP network bridging the infotainment and the safety critical domains, in: Embedded Security in Cars (ESCAR 2015 USA.

[52] F. Luo, S. Hou, Security mechanisms design of automotive gateway firewall, in: SAE International, 2019, http://dx.doi.org/10.4271/2019-01-0481.

[53] AUTOSAR, Specification of Secure Onboard Communication, AUTOSAR.

[54] W. Busch, Boosting Security in cars with CAN-FD, Technical Report, Avnet Silica, 2018.

[55] AUTOSAR, Requirements on Crypto Stack, AUTOSAR.

[56] A. Berthold, M.P. Schneider, AUTOSAR Security: Achieving Integrated Cybersecurity with the Adaptive Platform, Technical Report, ESCRYPT, 2020.

[57] M. Rumez, D. Grimm, R. Kriesten, E. Sax, An overview of automotive service-oriented architectures and implications for security countermeasures, IEEE Access 8 (2020) 221852–221870, http://dx.doi.org/10.1109/ACCESS.2020.3043070.

[58] AUTOSAR, Explanation of IPsec: Implementation Guidelines, AUTOSAR.

[59] AUTOSAR, Specification of Identity and Access Management, AUTOSAR.

[60] R. Buttigieg, M. Farrugia, C. Meli, Security issues in controller area networks in automobiles, in: 2017 18th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA), 2017, pp. 93–98, http://dx.doi.org/10.1109/STA.2017.8314877.

[61] S. Abbott-McCune, L.A. Shay, Techniques in hacking and simulating a modem automotive controller area network, in: 2016 IEEE International Carnahan Conference on Security Technology (ICCST), 2016, pp. 1–7, http://dx.doi.org/10.1109/CCST.2016.7815712.

[62] J. Valldorf, W. Gessner (Eds.), Embedded security solutions for automotive applications, in: Advanced Microsystems for Automotive Applications 2007, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 177–191, http://dx.doi.org/10.1007/978-3-540-71325-8_14.

[63] M. Kang, J. Kang, A novel intrusion detection method using deep neural network for in-vehicle network security, in: 2016 IEEE 83rd Vehicular Technology Conference (VTC Spring), 2016, pp. 1–5, http://dx.doi.org/10.1109/VTCSpring.2016.7504089.

[64] A. Taylor, S. Leblanc, N. Japkowicz, Anomaly detection in automobile control network data with long short-term memory networks, in: 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), 2016, pp. 130–139, http://dx.doi.org/10.1109/DSAA.2016.20.

[65] G. Loukas, T. Vuong, R. Heartfield, G. Sakellari, Y. Yoon, D. Gan, Cloud-based cyber-physical intrusion detection for vehicles using deep learning, IEEE Access 6 (2018) 3491–3508, http://dx.doi.org/10.1109/ACCESS.2017.2782159.

[66] J. Wang, H. Zhang, F. li, Z. Wang, J. Zhao, Intelligent vehicle knowledge representation and anomaly detection using neural knowledge DNA, J. Inf. Secur. Appl. 52 (2020) 102498, http://dx.doi.org/10.1016/j.jisa.2020.102498, URL: http://www.sciencedirect.com/science/article/pii/S2214212618307282.

[67] S. Boumiza, R. Braham, An anomaly detector for CAN bus networks in autonomous cars based on neural networks, in: 2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2019, pp. 1–6, http://dx.doi.org/10.1109/WiMOB.2019.8923315.

[68] H.M. Song, J. Woo, H.K. Kim, In-vehicle network intrusion detection using deep convolutional neural network, Veh. Commun. 21 (2020) 100198, http://dx.doi.org/10.1016/j.vehcom.2019.100198, URL: http://www.sciencedirect.com/science/article/pii/S2214209619302451.

[69] J. Xiao, H. Wu, X. Li, Robust and self-evolving IDS for in-vehicle network by enabling spatiotemporal information, in: 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2019, pp. 1390–1397, http://dx.doi.org/10.1109/HPCC/SmartCity/DSS.2019.00193.

[70] M. Hanselmann, T. Strauss, K. Dormann, H. Ulmer, CANet: An unsupervised intrusion detection system for high dimensional CAN bus data, IEEE Access 8 (2020) 58194–58205, http://dx.doi.org/10.1109/access.2020.2982544.

[71] V.S. Barletta, D. Caivano, A. Nannavecchia, M. Scalera, A kohonen SOM architecture for intrusion detection on in-vehicle communication networks, Appl. Sci. 10 (15) (2020) http://dx.doi.org/10.3390/app10155062, URL: https://www.mdpi.com/2076-3417/10/15/5062.

[72] S.N. Narayanan, S. Mittal, A. Joshi, OBD SecureAlert: An anomaly detection system for vehicles, in: 2016 IEEE International Conference on Smart Computing (SMARTCOMP), 2016, pp. 1–6, http://dx.doi.org/10.1109/SMARTCOMP.2016.7501710.

[73] D.D. Abdulaziz Alshammari, G. Corser, Classification approach for intrusion detection in vehicle systems, Wirel. Eng. Technol. 1 (1) (2018) 79–94, http://dx.doi.org/10.4236/wet.2018.94007.

[74] D. Tian, Y. Li, Y. Wang, et al., An intrusion detection system based on machine learning for CAN-bus, in: Industrial Networks and Intelligent Systems, INISCOM 2017. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 221, 2018, pp. 285–294, http://dx.doi.org/10.1007/978-3-319-74176-5_25.

[75] C. Wang, Z. Zhao, L. Gong, L. Zhu, Z. Liu, X. Cheng, A distributed anomaly detection system for in-vehicle network using HTM, IEEE Access 6 (2018) 9091–9098.

[76] Y. Hamada, M. Inoue, N. Adachi, et al., Intrusion detection system for in-vehicle networks, SEI Tech. Rev. 1 (88) (2019) 76–81.

[77] M. Al-Saud, A.M. Eltamaly, M.A. Mohamed, A. Kavousi Fard, An intelligent data-driven model to secure intra-vehicle communications based on machine learning, IEEE Trans. Ind. Electron. (2019) 1, http://dx.doi.org/10.1109/TIE.2019.2924870.

[78] O. Avatefipour, A.S. Al-Sumaiti, A.M. El-Sherbeeny, E.M. Awwad, M.A. Elmeligy, M.A. Mohamed, H. Malik, An intelligent secured framework for cyberattack detection in electric vehicles' CAN bus using machine learning, IEEE Access 7 (2019) 127580–127592, http://dx.doi.org/10.1109/ACCESS.2019.2937576.

[79] C. Lin, A. Sangiovanni-Vincentelli, Cyber-security for the controller area network (CAN) communication protocol, in: 2012 International Conference on Cyber Security, 2012, pp. 1–7, http://dx.doi.org/10.1109/CyberSecurity.2012.7.

[80] Z. King, S. Yu, Investigating and securing communications in the controller area network (CAN), in: 2017 International Conference on Computing, Networking and Communications (ICNC), 2017, pp. 814–818, http://dx.doi.org/10.1109/ICCNC.2017.7876236.

[81] W.A. Farag, Cantrack: Enhancing automotive CAN bus security using intuitive encryption algorithms, in: 2017 7th International Conference on Modeling, Simulation, and Applied Optimization (ICMSAO), 2017, pp. 1–5, http://dx.doi.org/10.1109/ICMSAO.2017.7934878.

[82] P. Noureldeen, M.A. Azer, A. Refaat, M. Alam, Replay attack on lightweight CAN authentication protocol, in: 2017 12th International Conference on Computer Engineering and Systems (ICCES), 2017, pp. 600–606, http://dx.doi.org/10.1109/ICCES.2017.8275376.

[83] A.W. Kyusuk Han, K.G. Shin, Automotive cybersecurity for in-vehicle communication, IQT Quart. 6 (1) (2014) 22–25.

[84] H. Ueda, R. Kurachi, H. Takada, et al., Security authentication system for in-vehicle network, in: SEI Technical Review, (81) 2015, pp. 1–5.

[85] A. Tashiro, H. Muraoka, S. Araki, K. Kakizaki, S. Uehara, A secure protocol consisting of two different security-level message authentications over CAN, in: 2017 3rd IEEE International Conference on Computer and Communications (ICCC), 2017, pp. 1520–1524, http://dx.doi.org/10.1109/CompComm.2017.8322794.

[86] Z. Lu, Q. Wang, X. Chen, G. Qu, Y. Lyu, Z. Liu, LEAP: A lightweight encryption and authentication protocol for in-vehicle communications, in: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2019, pp. 1158–1164, http://dx.doi.org/10.1109/ITSC.2019.8917500.

[87] M. Zhang, A. Masrur, Improving timing behavior on encrypted CAN buses, in: 2019 IEEE 25th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), 2019, pp. 1–6, http://dx.doi.org/10.1109/RTCSA.2019.8864567.

[88] S. Fassak, Y. El Hajjaji El Idrissi, N. Zahid, M. Jedra, A secure protocol for session keys establishment between ECUs in the CAN bus, in: 2017 International Conference on Wireless Networks and Mobile Communications (WINCOM), 2017, pp. 1–6, http://dx.doi.org/10.1109/WINCOM.2017.8238149.

[89] A.S. Siddiqui, C. Lee, W. Che, J. Plusquellic, F. Saqib, Secure intra-vehicular communication over CANFD, in: 2017 Asian Hardware Oriented Security and Trust Symposium (AsianHOST), 2017, pp. 97–102, http://dx.doi.org/10.1109/AsianHOST.2017.8354002.

[90] T. Youn, Y. Lee, S. Woo, Practical sender authentication scheme for in-vehicle CAN with efficient key management, IEEE Access 8 (2020) 86836–86849.

[91] M.R. Ansari, S. Yu, Q. Yu, IntelliCAN: Attack-resilient controller area network (CAN) for secure automobiles, in: 2015 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS), 2015, pp. 233–236, http://dx.doi.org/10.1109/DFT.2015.7315168.

[92] Y. Gui, A.S. Siddiqui, F. Saqib, Hardware based root of trust for electronic control units, in: SoutheastCon 2018, 2018, pp. 1–7, http://dx.doi.org/10.1109/SECON.2018.8479266.

[93] S. Woo, D. Moon, T. Youn, Y. Lee, Y. Kim, CAN ID Shuffling technique (CIST): Moving target defense strategy for protecting in-vehicle CAN, IEEE Access 7 (2019) 15521–15536.

[94] H. Mun, K. Han, D.H. Lee, Ensuring safety and security in CAN-based automotive embedded systems: A combination of design optimization and secure communication, IEEE Trans. Veh. Technol. 69 (7) (2020) 7078–7091.

[95] B. Groza, L. Popa, P. Murvay, Highly efficient authentication for CAN by identifier reallocation with ordered CMACs, IEEE Trans. Veh. Technol. 69 (6) (2020) 6129–6140.

[96] T. Lenard, R. Bolboacă, B. Genge, P. Haller, MixCAN: Mixed and backward-compatible data authentication scheme for controller area networks, in: 2020 IFIP Networking Conference (Networking), 2020, pp. 395–403.

[97] B. Groza, P. Murvay, Efficient intrusion detection with bloom filtering in controller area networks, IEEE Trans. Inf. Forensics Secur. 14 (4) (2019) 1037–1051, http://dx.doi.org/10.1109/TIFS.2018.2869351.

[98] M. Gmiden, M.H. Gmiden, H. Trabelsi, An intrusion detection method for securing in-vehicle CAN bus, in: 2016 17th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA), 2016, pp. 176–180, http://dx.doi.org/10.1109/STA.2016.7952095.

[99] A.J. Brown, T.R. Andel, M. Yampolskiy, J.T. McDonald, CAN authorization using message priority bit-level access control, in: 2018 1st International Conference on Data Intelligence and Security (ICDIS), 2018, pp. 1–8, http://dx.doi.org/10.1109/ICDIS.2018.00008.

[100] Q. Wang, Z. Lu, G. Qu, An entropy analysis based intrusion detection system for controller area network in vehicles, in: 2018 31st IEEE International System-on-Chip Conference (SOCC), 2018, pp. 90–95.

[101] J. Ning, J. Wang, J. Liu, N. Kato, Attacker identification and intrusion detection for in-vehicle networks, IEEE Commun. Lett. 23 (11) (2019) 1927–1930.

[102] S. Ohira, A.K. Desta, T. Kitagawa, I. Arai, K. Fujikawa, Divider: Delay-time based sender identification in automotive networks, 2020, URL: arXiv:2008.10941.

[103] S. Katragadda, P.J. Darby, A. Roche, R. Gottumukkala, Detecting low-rate replay-based injection attacks on in-vehicle networks, IEEE Access 8 (2020) 54979–54993.
[104] S. Tariq, S. Lee, H.K. Kim, S.S. Woo, CAN-ADF: The controller area network attack detection framework, Comput. Secur. 94 (2020) 101857, http://dx.doi.org/10.1016/j.cose.2020.101857, URL: http://www.sciencedirect.com/science/article/pii/S0167404820301292.
[105] P. Murvay, B. Groza, TIDAL-CAN: Differential timing based intrusion detection and localization for controller area network, IEEE Access 8 (2020) 68895–68912.
[106] T. Matsumoto, M. Hata, M. Tanabe, K. Yoshioka, K. Oishi, A method of preventing unauthorized data transmission in controller area network, in: 2012 IEEE 75th Vehicular Technology Conference (VTC Spring), 2012, pp. 1–5, http://dx.doi.org/10.1109/VETECS.2012.6240294.
[107] H. Kwon, S. Lee, J. Choi, B. Chung, Mitigation mechanism against in-vehicle network intrusion by reconfiguring ECU and disabling attack packet, in: 2018 International Conference on Information Technology (InCIT), 2018, pp. 1–5.
[108] M. Tian, R. Jiang, C. Xing, H. Qu, Q. Lu, X. Zhou, Exploiting temperature-varied ECU fingerprints for source identification in in-vehicle network intrusion detection, in: 2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC), 2019, pp. 1–8, http://dx.doi.org/10.1109/IPCCC47392.2019.8958766.
[109] K. Cheng, Y. Bai, Y. Zhou, Y. Tang, D. Sanan, Y. Liu, CANeleon: Protecting CAN bus with frame ID chameleon, IEEE Trans. Veh. Technol. 69 (7) (2020) 7116–7130.
[110] J. Zhou, P. Joshi, H. Zeng, R. Li, BTMonitor: Bit-time-based intrusion detection and attacker identification in controller area network, ACM Trans. Embed. Comput. Syst. 18 (6) (2019) http://dx.doi.org/10.1145/3362034.
[111] H. Olufowobi, C. Young, J. Zambreno, G. Bloom, SAIDuCANT: Specification-based automotive intrusion detection using controller area network (CAN) timing, IEEE Trans. Veh. Technol. 69 (2) (2020) 1484–1494, http://dx.doi.org/10.1109/TVT.2019.2961344.
[112] B. Robert Bosch GmbH, CAN With flexible data-rate, 2012, Specification Version 1.0.
[113] M. Wolf, A. Weimerskirch, C. Paar, Security in automotive bus systems, in: Proceedings of the Workshop on Embedded Security in Cars (ESCAR)'04, 2004.
[114] J.M. Ernst, A.J. Michaels, LIN bus security analysis, in: IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society, 2018, pp. 2085–2090, http://dx.doi.org/10.1109/IECON.2018.8592744.
[115] J. Takahashi, Y. Aragane, T. Miyazawa, H. Fuji, H. Yamashita, K. Hayakawa, S. Ukai, H. Hayakawa, Automotive attacks and countermeasures on LIN-bus, J. Inf. Process. 25 (2017) 220–228, http://dx.doi.org/10.2197/ipsjjip.25.220.
[116] J. Deng, L. Yu, Y. Fu, O. Hambolu, R.R. Brooks, Chapter 6 - Security and data privacy of modern automobiles, in: M. Chowdhury, A. Apon, K. Dey (Eds.), Data Analytics for Intelligent Transportation Systems, Elsevier, 2017, pp. 131–163, http://dx.doi.org/10.1016/B978-0-12-809715-1.00006-7, URL: http://www.sciencedirect.com/science/article/pii/B9780128097151000067.
[117] NI, FlexRay Automotive communication bus overview, in: National Instruments, vol. 1, 2019, pp. 1–7.
[118] FlexRay Consortium, FlexRay communications system, protocol specification, version 2.0, FlexRay TM 1 (1) (2010) 1–341.
[119] Y.-J.W. Meng-Zhuo Liu, Y.-N. Xu, Research of authenticated encryption security protocol for FlexRay in-vehicle network, Int. J. Comput. Theory Eng. 10 (5) (2018) 175–179, http://dx.doi.org/10.7763/IJCTE.2018.V10.1221.
[120] Z. chao Liu, Y. Wang, LM algorithm neural network predictive control of FlexRay bus system, J. Phys. Conf. Ser. 1267 (2019) 012094, http://dx.doi.org/10.1088/1742-6596/1267/1/012094.
[121] L. Huan, L. Chao, FlexRay Vehicle network predictive control based on neural network, MATEC Web Conf. 232 (2018) 01042, http://dx.doi.org/10.1051/matecconf/201823201042.
[122] X. He, Q. Wang, Z. Zhang, A survey of study of FlexRay systems for automotive net, in: Proceedings of 2011 International Conference on Electronic Mechanical Engineering and Information Technology, 3, 2011, pp. 1197–1204, http://dx.doi.org/10.1109/EMEIT.2011.6023309.
[123] A.R. Mousa, P. NourElDeen, M. Azer, M. Allam, Lightweight authentication protocol deployment over FlexRay, in: Proceedings of the 10th International Conference on Informatics and Systems, INFOS '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 233–239, http://dx.doi.org/10.1145/2908446.2908485.
[124] P.-S. Murvay, L. Popa, B. Groza, Accommodating time-triggered authentication to FlexRay demands, in: Proceedings of the Third Central European Cybersecurity Conference, CECC 2019, Association for Computing Machinery, New York, NY, USA, 2019, http://dx.doi.org/10.1145/3360664.3360666.
[125] Z. Gu, G. Han, H. Zeng, Q. Zhao, Security-aware mapping and scheduling with hardware co-processors for FlexRay-based distributed embedded systems, IEEE Trans. Parallel Distrib. Syst. 27 (10) (2016) 3044–3057.
[126] R. Radhiga, J. Pradeep, Design of FlexRay communication controller protocol for an automotive application, in: 2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO), 2015, pp. 1–7, http://dx.doi.org/10.1109/ISCO.2015.7282283.
[127] S. Shreejith, S.A. Fahmy, Extensible FlexRay communication controller for FPGA-based automotive systems, IEEE Trans. Veh. Technol. 64 (2) (2015) 453–465, http://dx.doi.org/10.1109/TVT.2014.2324532.
[128] T. Lee, C. Kuo, I. Lin, High performance CAN/FlexRay gateway design for in-vehicle network, in: 2017 IEEE Conference on Dependable and Secure Computing, 2017, pp. 240–242, http://dx.doi.org/10.1109/DESEC.2017.8073848.
[129] S. Shreejith, S.A. Fahmy, Extensible FlexRay communication controller for FPGA-based automotive systems, IEEE Trans. Veh. Technol. 64 (2) (2015) 453–465, http://dx.doi.org/10.1109/TVT.2014.2324532.
[130] D. Püllen, N.A. Anagnostopoulos, T. Arul, S. Katzenbeisser, Security and safety co-engineering of the FlexRay bus in vehicular networks, in: Proceedings of the International Conference on Omni-Layer Intelligent Systems, COINS '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 31–37, http://dx.doi.org/10.1145/3312614.3312626.
[131] D. Püllen, N.A. Anagnostopoulos, T. Arul, S. Katzenbeisser, Securing FlexRay-based in-vehicle networks, Microprocess. Microsyst. 77 (2020) 103144, http://dx.doi.org/10.1016/j.micpro.2020.103144, URL: http://www.sciencedirect.com/science/article/pii/S0141933120303112.
[132] F.B. Steffen Abbenseth, et al., Most the automotive multimedia network, in: Electronics Library MOST, vol. 1, 2011, pp. 1–5.
[133] S.-Y. Lee, S.-H. Park, H.-S. Choi, C.D. Lee, MOST Network system supporting full-duplexing communication, in: 2012 14th International Conference on Advanced Communication Technology (ICACT), 2012, pp. 1272–1275.
[134] M. Lee, S. Chung, H. Jin, Automotive network gateway to control electronic units through most network, in: 2010 Digest of Technical Papers International Conference on Consumer Electronics (ICCE), 2010, pp. 309–310, http://dx.doi.org/10.1109/ICCE.2010.5418726.
[135] A. Sumorek, M. Buczaj, New elements in vehicle communication media oriented systems transport protocol, TEKA. Comm. Mot. Energ. Agric. 12 (1) (2012) 275–279.
[136] S. Lee, B. Cho, Y. Choi, K. Baek, Implementation of MOST/CAN network protocol, in: 2011 International Conference on Electrical and Control Engineering, 2011, pp. 5974–5977, http://dx.doi.org/10.1109/ICECENG.2011.6057339.
[137] Z. Dong, Z. Piao, I. Jang, J. Chung, C. Lee, Design of FlexRay-MOST gateway using static segments and control messages, in: 2012 IEEE International Symposium on Circuits and Systems (ISCAS), 2012, pp. 536–539, http://dx.doi.org/10.1109/ISCAS.2012.6272085.
[138] M. Lee, S. Chung, H. Jin, Automotive network gateway to control electronic units through most network, in: 2010 Digest of Technical Papers International Conference on Consumer Electronics (ICCE), 2010, pp. 309–310, http://dx.doi.org/10.1109/ICCE.2010.5418726.
[139] C. Varun, M. Kathiresh, Automotive ethernet in on-board diagnosis (over IP) in-vehicle networking, in: 2014 International Conference on Embedded Systems (ICES), 2014, pp. 255–260, http://dx.doi.org/10.1109/EmbeddedSys.2014.6953168.
[140] R.M. Daoud, H.H. Amer, H.M. Elsayed, Y. Sallez, Ethernet-based car control network, in: 2006 Canadian Conference on Electrical and Computer Engineering, 2006, pp. 1031–1034, http://dx.doi.org/10.1109/CCECE.2006.277777.
[141] M. Postolache, G. Neamtu, S.D. Trofin, CAN - Ethernet gateway for automotive applications, in: 2013 17th International Conference on System Theory, Control and Computing (ICSTCC), 2013, pp. 422–427, http://dx.doi.org/10.1109/ICSTCC.2013.6688995.
[142] C. Bernardini, M.R. Asghar, B. Crispo, Security and privacy in vehicular communications: Challenges and opportunities, Veh. Commun. 10 (2017) http://dx.doi.org/10.1016/j.vehcom.2017.10.002.
[143] Open Alliance, BroadR-Reach, BroadR-Reach Specifications for Communication Channel. Technical Report.
[144] AUTOSAR, SOME/IP Protocol Specification, AUTOSAR.
[145] A. Mayr, M. Helmling, Middleware Protocols in the Automobile, Technical Report.
[146] T. Kiravuo, M. Sarela, J. Manner, A survey of ethernet LAN security, IEEE Commun. Surv. Tutor. 15 (3) (2013) 1477–1491, http://dx.doi.org/10.1109/SURV.2012.121112.00190.
[147] M. Rahmani, J. Hillebrand, W. Hintermaier, R. Bogenberger, E. Steinbach, A novel network architecture for in-vehicle audio and video communication, in: 2007 2nd IEEE/IFIP International Workshop on Broadband Convergence Networks, 2007, pp. 1–12, http://dx.doi.org/10.1109/BCN.2007.372741.
[148] S. Jeong, B. Jeon, B. Chung, H.K. Kim, Convolutional neural network-based intrusion detection system for avtp streams in automotive ethernet-based networks, Veh. Commun. 29 (2021) 100338, http://dx.doi.org/10.1016/j.vehcom.2021.100338, URL: https://www.sciencedirect.com/science/article/pii/S2214209621000073.

Alfonso Martínez-Cruz obtained the degree of Doctor in Computer Science at the Center for Research in Computation of the IPN in 2016, in Mexico City. During his doctoral studies, he did a research stay at the University of California at Santa Barbara (UCSB). He worked in the industry, in the design and development of HW-SW, he has published in National and International conferences and JCR journals. He currently works as researcher at National Institute of Astrophysics, Optics and Electronics. His lines of interest are: in-vehicle protocols, Security in IoT platforms, HW-SW architectures, Artificial intelligence, Optimization algorithms, Security in mobile devices, Design of HW-SW real time systems, and design of Intelligent systems.

Kelsey A. Ramírez-Gutiérrez received her M.S. degree in 2010, and her Ph.D. in 2014 from the Mechanical and Electrical Engineering School of the National Polytechnic Institute of México. She is a CONACyT researcher in Instituto Nacional de Astrofísica, Óptica y Electrónica, where she collaborates with the Cybersecurity Laboratory. Her research interests are in the fields of digital image processing and information security.

Claudia Feregrino-Uribe is a researcher at the Computer Science Department at INAOE, Puebla, Mexico. Her research areas are Cryptography, Watermarking, and Digital Systems Design. She received her B.Sc. in Computer Systems Engineering from Queretaro Institute of Technology, M.Sc. in Electrical Engineering with Telecommunications option from the CINVESTAV, Guadalajara, and Ph.D. from Electronic Engineering in Digital Systems from Loughborough University in the United Kingdom. Dr. Feregrino has published 100+ papers in scientific journals and international conferences, she is an associate editor for several international journals and has been involved in the organization or as a PC member for several conferences/workshops.

Alicia Morales-Reyes was admitted to the PhD degree in the College of Science and Engineering at the University of Edinburgh in 2011. She developed her research within the System Level Integration Group at the Institute for Integrated Micro and Nano Systems. In 2006, she received the MSc degree in Computer Science from the Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE) in Tonantzintla, Mexico. She obtained a BEng on Electrical and Electronics Engineering from the National Autonomous University of Mexico, in 2002. Currently, she is a titular researcher in the Computer Science department at INAOE. She collaborates within the reconfigurable and high-performance computing research group.