In-Vehicle Networks

Subjects: Engineering, Electrical & Electronic

Contributor: Narayan Khatri

Modern vehicles are no longer simply mechanical devices. Connectivity between the vehicular network and the outside world has widened the security holes that hackers can use to exploit a vehicular network. Controller Area Network (CAN), FlexRay, and automotive Ethernet are popular protocols for in-vehicle networks (IVNs) and will stay in the industry for many more years. However, these protocols were not designed with security in mind. They have several vulnerabilities, such as lack of message authentication, lack of message encryption, and an ID-based arbitration mechanism for contention resolution. Adversaries can use these vulnerabilities to launch sophisticated attacks that may lead to loss of life and damage to property. Thus, the security of the vehicles should be handled carefully. In this paper, we investigate the security vulnerabilities with in-vehicle network protocols such as CAN, automotive Ethernet, and FlexRay. A comprehensive survey on security attacks launched against in-vehicle networks is presented along with countermeasures adopted by various researchers. Various algorithms have been proposed in the past for intrusion detection in IVNs. However, those approaches have several limitations that need special attention from the research community. Blockchain is a good approach to solving the existing security issues in IVNs, and we suggest a way to improve IVN security based on a hybrid blockchain.

in-vehicle networks | Controller Area Network (CAN) | automotive Ethernet | FlexRay | security | vehicle | blockchain | intrusion detection system (IDS) | machine learning

1. Introduction

Recent decades have seen significant advancements in technology for self-driving vehicles and smart cars. Vehicular networks are networks of vehicle nodes providing various facilities, such as traffic management, parking management, accident avoidance, critical message dissemination, etc. [1]. There are various research fields where these vehicle nodes act as a communication messenger, such as Vehicular Ad-hoc Networks (VANETs), the Internet of Vehicles (IoV), Vehicle-to-Everything (V2X) communications, etc. There is a separate research field for in-vehicle networks (IVNs), dealing with the connections between the Engine Control Unit (ECU), the Transmission Control Unit (TCU), the Anti-lock Braking System (ABS), Body Control Modules (BCMs), and the various sensors inside the vehicle. There are protocols like Controller Area Network (CAN), FlexRay, and Ethernet, which help in the smooth functioning of in-vehicle networks [2].

The automotive industry is moving toward autonomous and connected vehicles. There are various benefits to going through this option. First of all, connected vehicles provide convenience and flexibility to passengers. Another benefit is that, with the inter-connections between vehicles, more information is shared regarding safety and

security, which leads to more secure transportation systems. Up to now, VANETs have mainly focused on providing efficiency and safety for drivers and passengers in a vehicular network. Today, Connected and Autonomous Electric Vehicles (CAEVs) is a new technology that is booming [3]. The main idea is that vehicles can interact with the external world through smart devices. This leads to various security issues as vehicles connect to the internet [3]. The main goal of the VANET is to provide vehicles with critical information (such as accident reports) with guaranteed performance in terms of latency and accuracy. However, this is a challenging task, as discussed in [4]. The increase in connectivity among transportation facilities, and the advancements in technologies, such as V2X-communications, have expanded the security holes, through which attackers can break into an in-vehicle network. Researchers have shown that it is possible to hack into a vehicle (either physically or remotely) through telematics systems, the on-board diagnostics (OBD-II) port, infotainment systems, and so on. Miller and Valasek were able to remotely kill a jeep on the highway by accessing its in-vehicle networks via the CAN-bus [5][6]. There are various researchers who focus on the drawbacks of the in-vehicle network protocol design that may allow a hacker access to in-vehicle networks.

The CAN protocol is widely used for in-vehicle communications in controlling and providing various functionalities to a vehicular system. It helps critical real-time communications for engine management, brake control, airbags, body systems control, etc. The CAN protocol helps in the transmission of CAN packets between various ECUs via inter-connected buses inside a vehicle network. Initially, CAN was used in automobiles owing to its simplicity, deterministic contention resolution mechanism, reduced network complexity, and low wiring costs. However, it was not designed with security in mind. CAN is based on broadcast communications, i.e., every ECU connected to the bus can send/receive messages on the bus. There is no authentication mechanism for messages, and encryption is not applied. Furthermore, the identifier (ID)-based priority mechanism for arbitration processes makes the CAN vulnerable to attack [7]. Hackers can mount attacks to disable functionalities in the vehicle, either locally by using an OBD-II diagnostics tool or through remote access to the telematics or infotainment systems. Thus, the security of in-vehicle CAN networks is vital to the overall security in modern transportation systems. This paper provides a comprehensive state-of-the-art survey of various security attacks that are mounted against in-vehicle network protocols. We also explore intrusion detection systems (IDSs) and other security solutions that were developed as countermeasures to those attacks. Furthermore, we provide security solutions for in-vehicle networks by using a hybrid blockchain framework.

Blockchain is a secured framework that is resistant to data modification. It has features like decentralization (i.e., no need to rely on a centralized node for recording, storing, and updating data), transparency (data records are visible to all nodes in the blockchain network), and it is open-source (i.e., open to the public), autonomous (i.e., nodes can update or transfer data by themselves), immutable (i.e., records are not corrupted once they become part of the chain), and anonymous (i.e., transactions are private) [8][9].

2. In-Vehicle Networks

In-vehicle networks are specialized internal communication networks that interconnect various components inside a vehicle. The components inside the vehicle include ECUs, gateways, sensors, actuators, etc. It is estimated that

modern high-tech cars have up to 70 ECUs with 2500 electronic signals being exchanged among the various components [10][11]. There are various types of electronic control units, such as the ECU, the TCU, the ABS, BCMs, the Speed Control Unit, the Battery Management System (BMS), the Powertrain Control Module, and the Door Control Unit (DCU). These electronic units receive input from various sensors for computations. Sensors are incorporated into the vehicle to help recognize and solve possible problems, including needing repair, servicing, etc. Sensors play a key role in automobiles. Typical functions include monitoring the crankshaft's rotation speed, managing the car speed, verifying the speed of the wheels, checking fuel temperature, monitoring tire pressure, monitoring the exhaust gases to check the oxygen ratio, computing air density in the engine, etc. All these functionalities provided by various sensors help drivers with early problem detection and accident prevention. The electronic control modules exchange data during normal operation of the vehicle. For instance, the ECU provides information about engine speed to the TCU, and the TCU tells other components when a gear shift will take place. Each unit has its predefined function and controls specific components using a standard protocol over the vehicle network. The protocols used for in-vehicle networks include CAN, Local Interconnect Network (LIN), FlexRay, Ethernet, and Media Oriented Systems Transport (MOST) [2]. These protocols are specifically designed to transmit messages within a pre-defined time limit with assurances on message delivery. Table 1 compares various in-vehicle networks in terms of bandwidth, application domain, advantages and disadvantages.

Table 1. Comparison of various pfrotocols used in in-vehicle network communications.

Protocol	Bandwidth	Application Domain	Advantages	Disadvantages
CAN	125 Kbps–1 Mbps	Widely used in powertrain and body control domains	Low cost, no need of central coordinator	Less bandwidth
LIN	1 Kbps–20 Kbps	Widely used in simple and less time-critical applications	Low cost, easy to implement	Low speed
FlexRay	10 Mbps	Widely used in advanced chassis control	High speed, better fault tolerance than CAN and Lin	High cost
MOST	24 Mbps	Widely used in infotainment applications	High speed	High cost
ETHERNET	100 Mbps	Widely used in the future in applications requiring high bandwidths	High speed (100 times faster than CAN bus)	High cost per node

The in-vehicle network architecture can be of the classic Electrical and Electronics (E/E) type with a central gateway, or a domain-based E/E type with several functional domains connected through a central gateway [12]. The communication overhead is increasing in the classical E/E architecture, because a variety of ECUs have to communicate and route data through a central gateway. In order to reduce this bottleneck, a domain-based architecture was developed, where various functional domains with a domain controller are interconnected through the central gateway. Here, most of the data communication occurs within these functional domains, reducing the

communication load on the gateway. Furthermore, the architecture is scalable to allow adding more functional domains.

2.1. The Controller Area Network Protocol

CAN is one of the well-known vehicle bus standards for in-vehicle networks. It is popular in automotive and industrial applications due to its low cost and flexible design, thereby reducing the wiring harness. For example, the number of wires was reduced by 40% in the Peugeot 307, which embeds two CAN buses [13]. CAN is a message-based protocol; the packets do not have information about the sender and receiver of the messages, and every node can read the messages transmitted over the bus. Functions supported by the protocol in the automotive domain include auto start/stop, electric parking brakes, parking assistance, automatic lane detection, collision avoidance, etc. Figure 1 shows a CAN bus node. It consists of the central processing unit (CPU), the CAN controller, and a transceiver. The function of the CPU is to decode the received messages and to decide on the messages to transmit. Each node can send or receive messages, but not simultaneously. A message or frame consists of an ID and a data payload of up to eight bytes (64-bits).

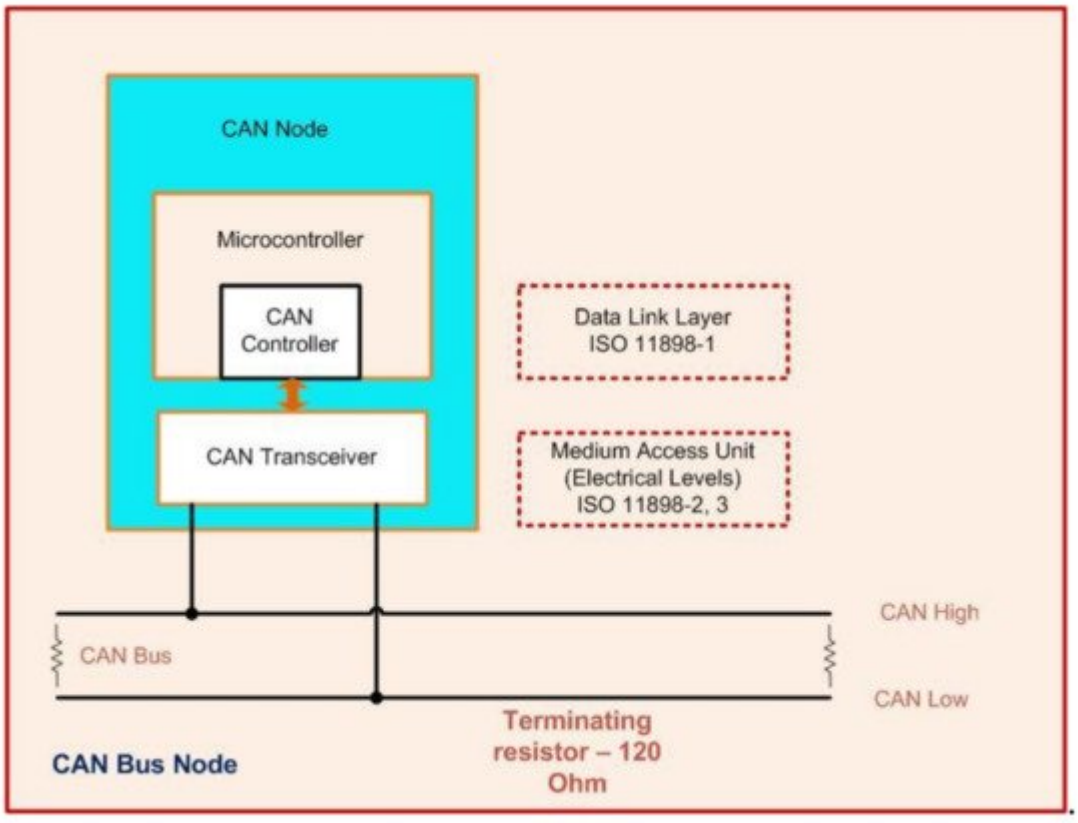

Figure 1. Controller area network bus node.

The CAN standard frame format consists of an 11-bit identifier. The identifier of the CAN frame represents the message priority. If a message has a lower identifier value, it will have a high priority on the bus. This field is used in the arbitration process to avoid conflict when two nodes or more than two nodes are transmitting the messages

simultaneously on the bus. Figure 2 shows the CAN frame format. It consists of seven fields: Start of Frame (SOF), arbitration, control, data, Cyclical Redundancy Check (CRC), Acknowledge (ACK), and End of Frame (EOF).

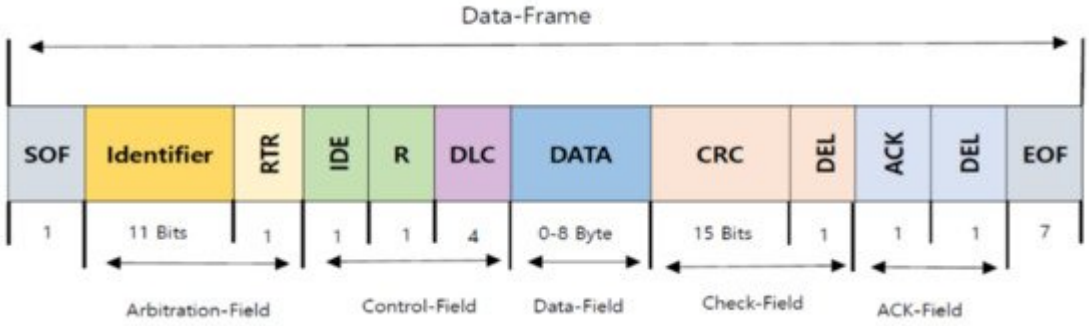

Figure 2. The controller area network (CAN) frame format (11-bit identifier).

CAN has four different frame types [14][15]. They are the data frame, the remote frame, the error frame, and the overload frame. The data frame is used for actual data transmission from a transmitter to other nodes (receivers). The remote frame is used by a node to request a certain message with a particular identifier. If any of the nodes on the bus detect an error, that node will transmit an error frame. The overload frame is used to inject a delay between the data and remote frames.

CAN Bus Attack Interfaces

The attack surface in the connected car environment was demonstrated by Aliwa et al. [16]. It consists of telematics units, infotainment systems, the OBD-II port, and sensors. Figure 3 illustrates the CAN bus attack surfaces. The attacker may inject messages into the network through a direct connection, like an OBD-II connection, or wirelessly through telematics or infotainment systems.

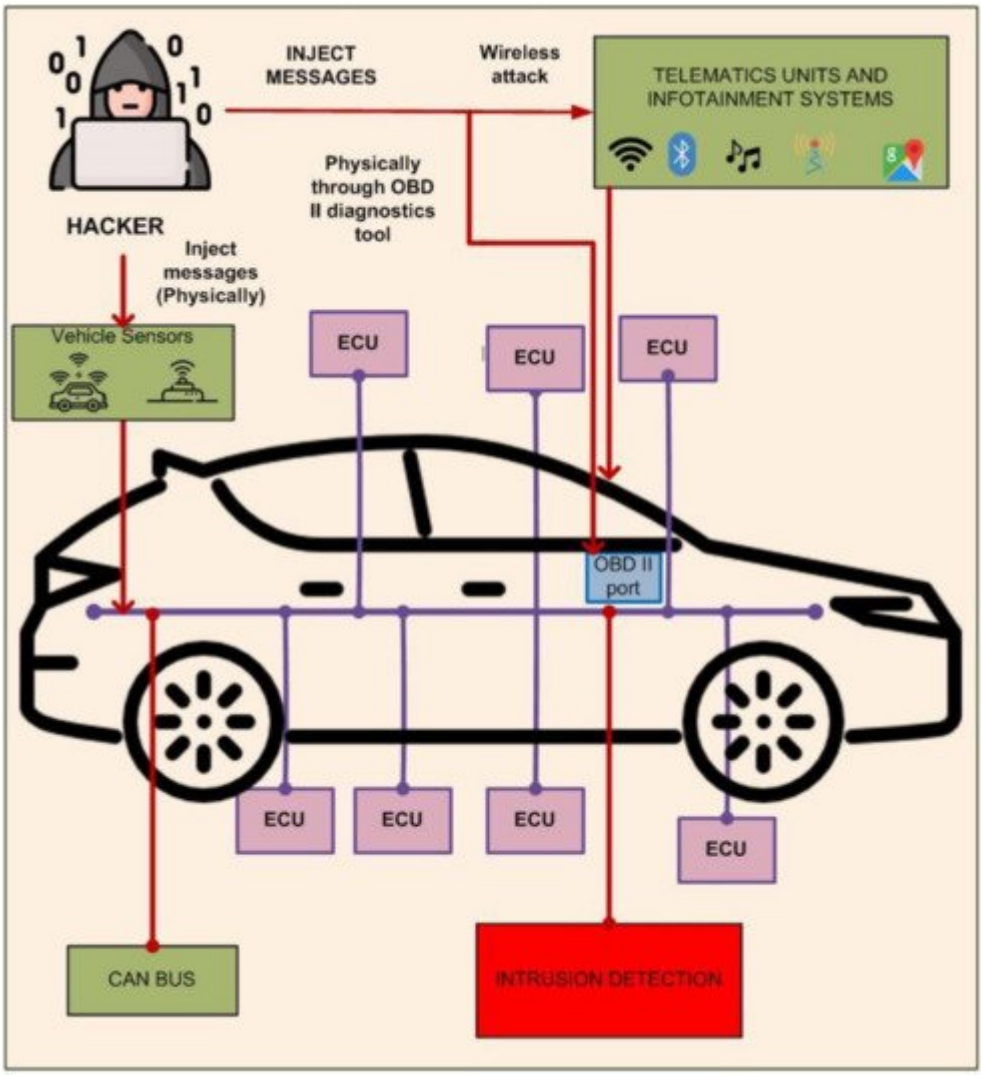

Figure 3. CAN bus attack interfaces.

2.2. Automotive Ethernet Protocol

The high-bandwidth requirements of modern vehicle applications motivated the introduction of automotive Ethernet (AE) as an essential component of in-vehicle networks [17]. Automotive Ethernet is going to be the future in-vehicle network protocol that satisfies the bandwidth requirements for multimedia applications, autonomous driving, and safety applications, such as the advanced driver-assistance system (ADAS) [18][19]. Other features of this protocol include efficient communication, lower latency, scalability, reduced wiring harness, and low cost. The various protocols used for AE are 100-Base-TX, BroadR-Reach (100Base-T1), Audio Video Bridge-Time-Sensitive Network (AVB/TSN), Diagnostics over Internet Protocol (DoIP), and Scalable Service-Oriented Middleware over Internet Protocol (SOME/IP), etc. [20]. 100Base-T1 is an AE standard that provides bandwidth of up to 100 Mbps over a single unshielded twisted-pair cable [19]. 1000Base-T1 is in development and will support data rates of up to 1000 Mbps [21]. Once a vehicle is on the market, the in-vehicle network protocols cannot be updated. Thus, the use of protocols and specifications should be handled mindfully during the design process. Since bus protocols are insecure, this emerging technology has room for enhanced security features. The security features that have to be

implemented in AE networks are access control to the network, secure on-board communication, a data access policy, anomaly detection and prevention mechanisms, etc. [18]. There is no authentication mechanism to enforce access control against unknown devices in AE networks to protect the various software and hardware components, such as the operating system, drivers, IP stack, Ethernet interfaces, the CPU, etc. [21]. There is a need for trust among devices participating in the network, which can be guaranteed through authentication mechanisms, as in IEEE 802.1x. The topology for in-vehicle network communication is a hierarchical structure where various protocols are connected to a central gateway [21]. However, this topology is neither dynamic nor extensible. If a hacker gets access to one port of the in-vehicle network, there is a chance he/she will also get access to other parts of the network that are connected through gateways. Electric Vehicles (EVs) are connected to the outside world through V2X communications, which uses Ethernet as a backbone [22], e.g., the communication between an EV and a charging station during negotiations for the charging service. There is an issue regarding the authenticity of the communicating bodies in vehicular networks. There should be a mechanism to evaluate authenticity and integrity, and protect the confidentiality of messages flowing between in-vehicle networks and outside networks [23]. The AUTOSAR consortium has not covered the issue of initializing pre-shared secret keys between senders and receivers of messages in Secure Onboard Communication (SecOC) [24]. Ethernet technology has been well studied in the cybersecurity community. Hackers can use previous knowledge from the IT domain and can launch sophisticated attacks against vehicle systems. Thus, future AE technology should be designed with these considerations in mind. Furthermore, cybersecurity attacks and their countermeasures should be investigated extensively in order to secure in-vehicle networks.

2.3. FlexRay Protocol

FlexRay is a reliable, time-triggered protocol that provides a higher bandwidth of up to 10 Mbps, compared to CAN networks, which provide data rates of up to 1 Mbps. All the ECUs connected by this protocol are synchronized to global time, and the data frames are transmitted and received within pre-defined time slots. This protocol has high fault-tolerance, compared to CAN. The properties of the FlexRay protocol regarding transmission capabilities include large payloads, flexibility in terms of network topologies, and the transmission of deterministic, as well as dynamic, data in one cycle. However, they have the drawbacks of having high cost and increased complexity in in-vehicle networks. The FlexRay frame consists of a header segment, a payload, and trailer segments. The header consists of the slot ID, payload length, cycle counter, etc. The payload segment contains the data. The trailer provides a CRC for the frame. This protocol is particularly used in drivetrain and chassis applications with time-critical and event-triggered messages. The vulnerabilities in this protocol are that it lacks confidentiality, authentication, and data freshness mechanisms [25]. The protocol is not designed to guarantee security from external attack. According to Shrestha and Kim [19], it is vulnerable to various attacks, such as eavesdropping, masquerading, injection, and replay attacks. Nilsson and Larson [26] mentioned that the application layer is missing, making it insecure. The CRC section of the FlexRay frame can protect the integrity of the data against transmission errors. Availability is guaranteed through time division multiplexing. These features help in providing safety to the network, but do not guarantee protection against security attacks.

References

1. Hartenstein, H.; Laberteaux, K.P. VANET: Vehicular Applications and Inter-Networking Technologies; John Wiley & Sons: Chichester, UK, 2009.
2. Zeng, W.; Khalid, M.A.S.; Chowdhury, S. In-Vehicle Networks Outlook: Achievements and Challenges. IEEE Commun. Surv. Tutorials 2016, 18, 1552–1571.
3. Rathee, G.; Sharma, A.; Iqbal, R.; Aloqaily, M.; Jaglan, N.; Kumar, R. A Blockchain Framework for Securing Connected and Autonomous Vehicles. Sensors 2019, 19, 3165.
4. Shrestha, R.; Nam, S.Y. Regional Blockchain for Vehicular Networks to Prevent 51% Attacks. IEEE Access 2019, 7, 95033–95045.
5. Greenberg, A. Hackers Remotely Kill a Jeep on the Highway-with Me in It. Available online: (accessed on 21 July 2015).
6. Kim, S.; Shrestha, R. Security and Privacy in Intelligent Autonomous Vehicles. In Automotive Cyber Security; J.B. Metzler: Stuttgart, Germany, 2020; pp. 35–66.
7. Lokman, S.-F.; Othman, A.T.; Abu-Bakar, M.-H. Intrusion detection system for automotive Controller Area Network (CAN) bus system: A review. EURASIP J. Wirel. Commun. Netw. 2019, 2019.
8. Lin, I.-C.; Liao, T.-C. A survey of blockchain security issues and challenges. Int. J. Netw. Sec. 2017, 19, 653–659.
9. Shrestha, R.; Nam, S.Y.; Bajracharya, R.; Kim, S. Evolution of V2X Communication and Integration of Blockchain for Security Enhancements. Electronics 2020, 9, 1338.
10. Kimm, H.; Ham, H.-S. Integrated Fault Tolerant System for Automotive Bus Networks. In Proceedings of the 2010 Second International Conference on Computer Engineering and Applications; Institute of Electrical and Electronics Engineers (IEEE), Bali, Indonesia, 19–21 March 2010; Volume 1, pp. 486–490.
11. Nilsson, D.; Larson, U.; Phung, P. Vehicle ECU classification based on safety-security characteristics. In Proceedings of the IET Road Transport Information and Control Conference and the ITS United Kingdom Members' Conference (RTIC 2008), Manchester, UK, 20–22 May 2008; p. 102.
12. Brunner, S.; Roder, J.; Kucera, M.; Waas, T. Automotive E/E-architecture enhancements by usage of ethernet TSN. In Proceedings of the 2017 13th Workshop on Intelligent Solutions in Embedded Systems (WISES), Hamburg, Germany, 12–13 June 2017; pp. 9–13.
13. Navet, N.; Simonot-Lion, F.; Delong, C. In-Vehicle Communication Networks: A Historical Perspective and Review; Apple Academic Press: Palm Bay, FL, USA, 2017; pp. 50–51.

14. Lee, H.; Jeong, S.H.; Kim, H.K. OTIDS: A Novel Intrusion Detection System for In-vehicle Network by Using Remote Frame. In Proceedings of the 2017 15th Annual Conference on Privacy, Security and Trust (PST), Calgary, AB, Canada, 28–30 August 2017; pp. 57–5709.
15. Bosch. Can Specifications. 1991. Available online: (accessed on 5 January 2021).
16. Aliwa, E.; Rana, O.; Perera, C.; Burnap, P. Cyberattacks and Countermeasures for In-Vehicle Networks. ACM Comput. Surv. 2021, 54, 1–37.
17. Kim, S.; Shrestha, R. Intelligent Autonomous Vehicle. In Automotive Cyber Security; J.B. Metzler: Stuttgart, Germany, 2020; pp. 15–33.
18. Jeon, B.; Ju, H.; Jung, B.; Kim, K.; Lee, D. A Study on Traffic Characteristics for Anomaly Detection of Ethernet-based IVN. In Proceedings of the 2019 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 16–18 October 2019; pp. 951–953.
19. Kim, S.; Shrestha, R. In-Vehicle Communication and Cyber Security. In Automotive Cyber Security; J.B. Metzler: Stuttgart, Germany, 2020; pp. 67–96.
20. Talic, A. Security analysis of ethernet in cars. Master's Thesis, Department of Communication Systems, KTH Royal Institute of Technology, Stockholm, Sweden, 2017. Available online: (accessed on 15 January 2021).
21. Corbett, C.; Schoch, E.; Kargl, F.; Felix, P. Automotive ethernet: Security opportunity or challenge? Sicherheit 2016 - Sicherheit, Schutz und Zuverlässigkeit, Bonn, Gesellschaft für Informatik e.V. 2016, pp. 45–54. Available online: (accessed on 27 January 2021).
22. Hank, P.; Muller, S.; Vermesan, O.; Keybus, J.V.D. Automotive Ethernet: In-vehicle Networking and Smart Mobility. In Proceedings of the Design, Automation & Test in Europe Conference & Exhibition (DATE), Grenoble, France, 18–22 March 2013; Volume 2013, pp. 1735–1739.
23. Glass, M.; Herrscher, D.; Meier, H.; Piastowski, M.; Schoo, P. "Seis"—Security in Embedded IP-Based Systems; ATZ Elektron Worldw: Berlin/Heidelberg, Germany, 2010; Volume 5, pp. 36–40.
24. Lee, H.-Y.; Lee, D.-H. Security of Ethernet in Automotive Electric/Electronic Architectures. J. Inst. Internet Broadcast. Commun. 2016, 16, 39–48.
25. Le, V.H.; Hartog, J.D.; Zannone, N. Security and privacy for innovative automotive applications: A survey. Comput. Commun. 2018, 132, 17–41.
26. Nilsson, D.K.; Larson, U.E.; Picasso, F.; Jonsson, E. A First Simulation of Attacks in the Automotive Network Communications Protocol FlexRay. In Proceedings of the Advances in Computer Science and Education; J.B. Metzler: Stuttgart, Germany, 2008; Volume 53, pp. 84–91.