In-Vehicle Network Attacks and Countermeasures: Challenges and Future Directions

Jiajia Liu, Shubin Zhang, Wen Sun, and Yongpeng Shi

Abstract

The emergence of in-vehicle networks, which are composed of controller area network (CAN) buses and a great number of ECUs, significantly reduces the difficulty of vehicle designing, repairing and refitting. While owing to the intrinsic vulnerabilities of in-vehicle networks and the increasingly rich interfaces to connect in-vehicle networks to the outside, adversarial attacks can be easily implemented on in-vehicle networks. Such attacks cause serious threats to automobile security and drivers' privacy. In light of this, we provide in this article a detailed guidance, explain the basic concepts, introduce the vulnerabilities of in-vehicle networks, and summarize the attacking methodologies. We also provide countermeasures for in-vehicle networks and point out challenges and future directions.

Introduction

Nowadays, modern automobiles have already become an indispensable part of daily life. Vehicular communication technologies, such as Vehicular Ad-hoc Network (VANET), connected vehicles and intelligent transportation systems (ITSs), have been proposed to meet the increasing data demand of vehicles [1]. In recent decades, the extensive use of electronic components makes modern automobiles no longer mere mechanical devices. A great number of electronic control units (ECUs) connected by several kinds of buses are used to control and monitor the state of vehicles [2]. The in-vehicle network, which is composed of buses and ECUs, reduces the difficulty of vehicle designing, repairing and refitting. People who have little knowledge about vehicles can diagnose the vehicle via a diagnostic trouble code (DTC) or improve the performance of vehicles by reprogramming corresponding firmware. Therefore, in-vehicle networks bring convenience to manufacturers, drivers and after-sales services.

Nevertheless, in-vehicle networks also bring up adversarial threats. Owing to several vulnerabilities of in-vehicle networks, adversaries can easily attack a target vehicle. Especially in recent years, providing connectivity between an automobile and the outside world has become the development tendency of modern automobiles. The existing interfaces, which are intended for connecting the external network, can be used by adversaries to access in-vehicle networks. If we fail to defend the in-vehicle network from attacks, adversaries can easily steal the driver's private information, or even take control of the target vehicle. We cannot expect the VANET and ITSs to continue to develop without ensuring vehicle security. In order to enhance in-vehicle network security, it is of great value to study the principle of the existing attacks and propose corresponding countermeasures.

In-vehicle network security has drawn extensive research attention. Studies on experimentally attacking in-vehicle networks have scored remarkable achievements. Hoppe and Dittman [3] implemented sniffing and replay attacks by using simulation of a simplified in-vehicle network, followed by [4] where the automotive components were used to implement a replay attack. Koscher *et al.* [5] first attacked two 2009 automobiles of the same model by physically accessing the in-vehicle network of the target car. They used a laptop that was connected with the OBD port to control a wide range of the automotive functions. Based on the previous study [5], Checkoway *et al.* [6] discussed the interfaces that can be used to access in-vehicle networks. In [7], a wireless attack on a midsize car was implemented. Woo *et al.* took control of the target vehicle with the help of malware installed on a smart phone.

In-vehicle network security is essential and these experimental studies are of great value. A growing number of researchers are interested in this field, while few tutorial works can be found in the literature. In light of this, we provide in this article some basic guidance to new researchers and a summary of available works. In particular, we introduce the basic concepts of in-vehicle networks, including the architecture of in-vehicle networks, and ECUs and controller area network (CAN) frame format. We summarize the vulnerabilities of in-vehicle networks in detail. A general procedure that can be followed to attack in-vehicle networks is provided, as a guide for new researchers in this field. According to most existing attacks, we introduce the methodologies that are efficient during the attacking process. We classify most existing countermeasures into three types, discuss the development tendency of in-vehicle networks, and point out research challenges and future directions.

The rest of this article is outlined as follows. In the following section, we introduce the architecture of in-vehicle networks, the CAN frame format and the vulnerabilities of in-vehicle networks. Then attacking methodologies and existing experimental studies are summarized. A general attacking procedure is also given. Following that, countermeasures for in-vehicle networks are discussed. Then challenges and future directions are presented. Finally, we summarize the article in the final section.

The authors are with Xidian University.

Digital Object Identifier: 10.1109/MNET.2017.1600257

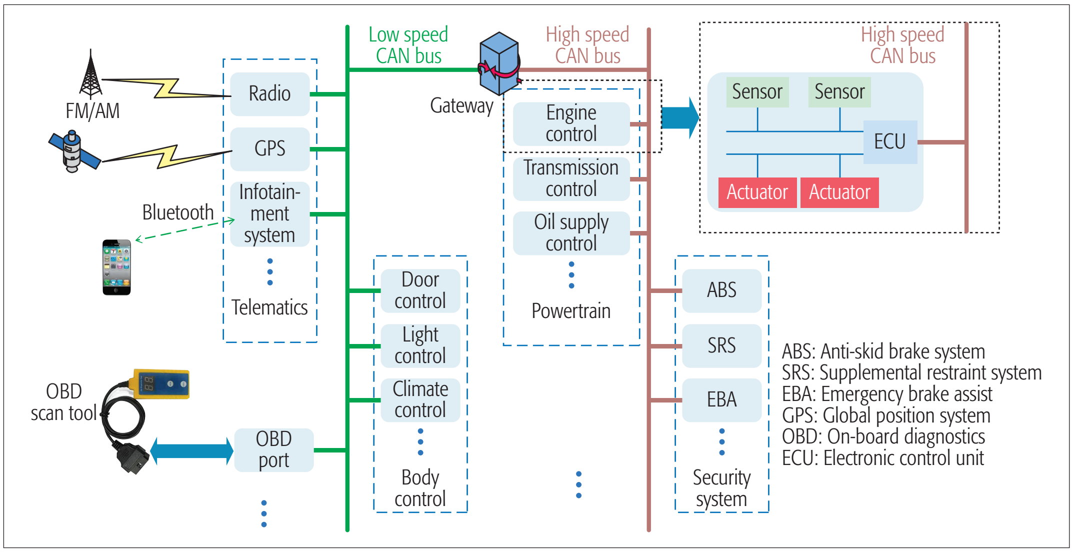

FIGURE 1. CAN network consists of high-speed CAN bus and low-speed CAN bus. The rate of high-speed CAN bus is about 500 kb/s. These two buses are bridged by a gate way. High-speed bus is used by the time-critical modules, while the low-speed bus is adopted by the less time-critical modules. For all modules, their ECUs are connected by CAN bus. An ECU, which has its own attached sensors and actuators, gets input from its sensors and implements specific functions by its actuators.

Vulnerabilities of In-Vehicle Networks

Since CAN buses have long been used in the majority of modern vehicles and most existing attacks on vehicle are based on CAN networks (i.e., an in-vehicle network adopting CAN bus technology), the in-vehicle network in this article mainly refers to a CAN network. In this section, the architecture of in-vehicle networks is introduced. Then the vulnerabilities of in-vehicle networks are summarized.

Architecture of In-Vehicle Networks

An in-vehicle network, consisting of ECUs and CAN buses, is shown in Fig. 1. ECUs are embedded devices designed to monitor vehicle state and take actions. Every ECU, which has its own attached sensors and actuators, gets input from its sensors and implements specific functions by its actuators. The ECUs, which have interactions with each other to implement specific functions, compose specific modules, such as the engine control module, body control module, instrument panel cluster and telematics module. Many actions require cooperation among different modules. For example, a simple braking needs complex interactions among the engine control module, antilock breaking system and instrument panel cluster.

The CAN protocol is a kind of bus protocol that is widely used in in-vehicle networks. Different ECUs are able to communicate with each other through the wired CAN bus. High-speed CAN buses and low-speed CAN buses are distinguished according to their data rate, and they are bridged by a gateway. The rate of a high-speed CAN bus is about 500 kb/s [8]. A high-speed CAN bus is adopted by the time-critical modules, such as the engine control module, brake control module and anti-skid brake system (ABS), while a low-speed CAN bus is usually used in less time-critical modules, including the instrument panel cluster, body control module, and telematics. The architecture of a CAN network makes it possible for an ECU to communicate with other ECUs, even if they are deployed on different-speed buses.

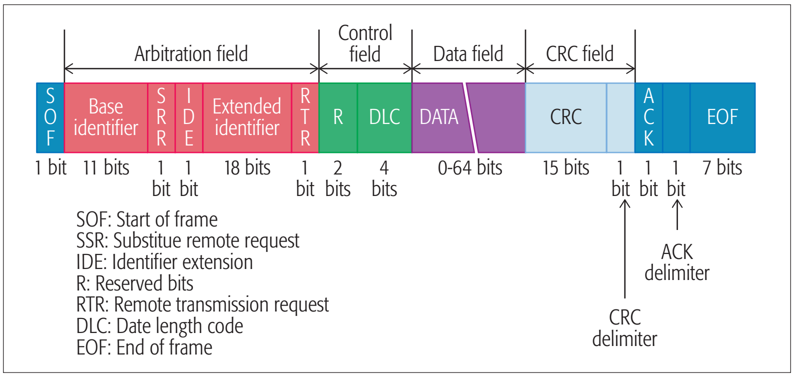

FIGURE 2. CAN data frames are divided into 2 types according to the length of arbitration field. Extended CAN data frame supports CAN 2.0B protocol, while base CAN data frame supports CAN 2.0A protocol. CAN 2.0B protocol are compatible with CAN 2.0A.

Vulnerabilities of CAN

CAN was developed in the early 1980s by Robert Bosch GmbH. Because of its high efficiency and low cost, the International Standardization Organization (ISO) established CAN as the international standard in 1993. The CAN protocol is extremely simple. Each ECU transmits messages to other ECUs using CAN frames. There are two types of identifiers in different CAN frames, extended identifiers and normal identifiers. Extended identifiers suit the CAN 2.0B protocol, while normal identifiers are applied to the CAN 2.0A protocol. Since CAN 2.0B supports compatibility with CAN 2.0A, only the CAN 2.0B frame (i.e., the frame with an extended identifier) is shown in Fig. 2. The CAN protocol has several intrinsic vulnerabilities, such as broadcast transmission, no authentication, no encryption, ID-based priority scheme and available interfaces. These vulnerabilities make in-vehicle networks vulnerable to malicious attacks.

Reference	Environment	Interface	Attacking methodologies	Contributions
Hoppe and Dittmann, 2007, [3]	Simulated software	CAN bus interface	•Frame sniffing •Replay attack	Implement frame sniffing and replay attack; control window lift.
Hoppe *et al.* 2008, [4]	Components from real car	CAN bus interface; OBD port	•Frame sniffing •Replay attack	Control the window lift, warning lights and air bag systems; stop these components from working.
Koscher *et al.*, 2010, [5]	Real car	OBD port	•Frame sniffing •Replay attack •DoS attack •Frame injection •Frame falsifying	Control body control module, radio, engine, and so on; implement DoS attack.
Checkoway *et al.*, 2011, [6]	Real car interfaces	•Indirect physical access (OBD port) •Short-range access (Bluetooth) •Long-range access (cellular)	Not applicable	Analyze attacking interfaces
Woo *et al.* 2015, [7]	Real car	OBD port (OBD scan tool)	•Frame sniffing •Replay attac •Frame injection •Frame falsifying	Implement wireless attack; control dash board, engine, handle control and so on.

TABLE 1. Experimental attacks on in-vehicle network.

> A large proportion of modern automobiles are equipped with a multifunctional telematics system, which supports the Global Position System, media entertainment, even directly accessing the cellular network. The ability to connect to external networks makes the telematics system easily vulnerable to cyber attacks, bringing potential security threats to in-vehicle networks.

Broadcast Transmission: CAN frames are broadcasted to all the nodes connected to the CAN bus. Every ECU receives the frames transmitted on the CAN bus and decide whether to take corresponding actions after identifying the frame ID. Due to the broadcast characteristic, malicious nodes can easily steal all the frames transmitted from other nodes.

No Authentication: As shown in Fig. 2, a CAN frame does not contain an authentication field or any other fields that can indicate its source. Hence, a receiver cannot distinguish the valid frames from the fake ones. This means a malicious node can disguise its identity and send fake frames to all the other nodes connected to the CAN bus. Adversaries can easily take control of the vehicle components by commanding the malicious nodes to send fake frames containing appropriate IDs to the CAN bus.

No Encryption: CAN frames are not encrypted. Adversaries can easily analyze the CAN frames based on a great number of historically recorded CAN frames.

ID-Based Priority Scheme: For a CAN frame, the identifier not only determines the target node but also indicates its priority. The smaller the identifier is, the higher the priority is. The transmission of a frame with high priority causes other frames to back off. For a simple example, if a malicious node is transmitting frames with the smallest identifier all the time, none of the legitimate nodes can transmit valid frames. It is obvious that this scheme makes in-vehicle networks vulnerable to denial of service (DoS) attacks.

Available Interfaces: For modern automobiles, there are several interfaces that can be used by adversaries to access in-vehicle networks, such as the OBD port, CD player, USB port and telematics systems (e.g., GM's OnStar, BMW's ConnectedDrive, Toyota's G-book). Among the mentioned interfaces, the OBD port is the most significant one, as it is designed for diagnosing the vehicle and reprogramming the ECUs' firmware. In recent years, most experimental attacks on vehicles were implemented based on the OBD port, since the OBD port has the ability to access the CAN bus and receive all the messages transmitted from other nodes (Table 1). A laptop connecting the OBD port can easily snoop the messages transmitted on the CAN bus. It should be noted that some OBD scan tools that are connected with a smart phone make wireless attack (i.e., via Bluetooth and WiFi) possible [7]. Another common interface is the telematics system. A large proportion of modern automobiles are equipped with a multifunctional telematics system, which supports the Global Position System (GPS), media entertainment, even directly accessing the cellular network. The ability to connect to external networks makes the telematics system easily vulnerable to cyber attacks, bringing potential security threats to in-vehicle networks. Adversaries are able to access the target in-vehicle network through the aforementioned interfaces, and implement various attacks, such as a replay attack, DoS attack, frame sniffing and frame injection.

In-Vehicle Network Attacks

In order to enhance in-vehicle network security, it is valuable to study the principle of in-vehicle network attacks. A series of experimental studies on how to attack in-vehicle networks have been accomplished. In this section, we make a conclusion on the methodologies that have been used in the experimental studies, present a general procedure that can be followed to attack in-vehicle networks, and introduce the existing experimental studies.

Attacking Methodologies

The methodologies that have been proved to be effective in the existing studies are summarized as follows.

Frame Sniffing: Frame sniffing is the foundation of other attacks. As mentioned earlier, CAN frames are broadcasted to all nodes. A malicious node can snoop all the frames transmitted on the CAN bus. By accessing an in-vehicle network via available interfaces (e.g., the OBD port, telematics system), adversaries can observe the traffic on the CAN bus and record a great number of valid frames. After analyzing historically recorded frames, the details of the CAN frames can be known. In [5], Koscher *et al*. found that the range of valid CAN frames is small. Hence, by iteratively testing CAN frames (i.e., the fuzzing test mentioned in [5]), adversaries are able to discover many functions of selected ECUs.

Frame Falsifying: Having known the details of the CAN frames, adversaries can design fake frames that will be sent to the in-vehicle network to implement specific attacks. These fake frames contain false data that can mislead corresponding legitimate ECUs. Adversaries are able to falsify the fuel level, change the speedometer reading and display failure information on the instrument panel cluster by falsifying frame data. The incorrect information may fool the driver, or even worse, cause dangerous behaviors.

Frame Injection: Adversaries can use a malicious node to send fake frames to the CAN bus. The malicious node could be a laptop connecting the OBD port, a reprogrammed ECU, or the telematics system infected by malware. In most existing studies, specific software is developed to inject fake frames to an in-vehicle network. The broadcast characteristic makes fake frames available to all nodes. Setting appropriate frames' ID makes the target node on the CAN bus accept these fake frames.

Replay Attack: A replay attack is a kind of simple attack. Adversaries only need to command the malicious nodes to send valid frames to the CAN bus at the appropriate time. Due to no authentication scheme in the CAN protocol, ECUs cannot identify whether the source of the received frames is legitimate or not. Although a replay attack can be easily implemented, the threats brought by a replay attack are non-negligible. For a stationary car, adversaries are able to open the door, start the engine, turn on the lights and drive the car away by injecting valid CAN frames to the car's in-vehicle network. Hoppe and Dittmann [3] implemented a replay attack in a simulation environment. Koscher *et al.* in [5] proved that the replay attack can be implemented in a real car scenario.

DoS Attack: A CAN network is vulnerable to DoS attacks. As mentioned earlier, the frame's ID indicates the priority of the frame. If there exist frames with the highest priority that are being transmitted on the CAN bus, no node is allowed to send frames to the in-vehicle network. Obviously, adversaries can easily implement DoS attacks by commanding the malicious node to send frames with the highest priority all the time. In [5], the authors disabled the communication of individual components on the CAN bus via a DoS attack.

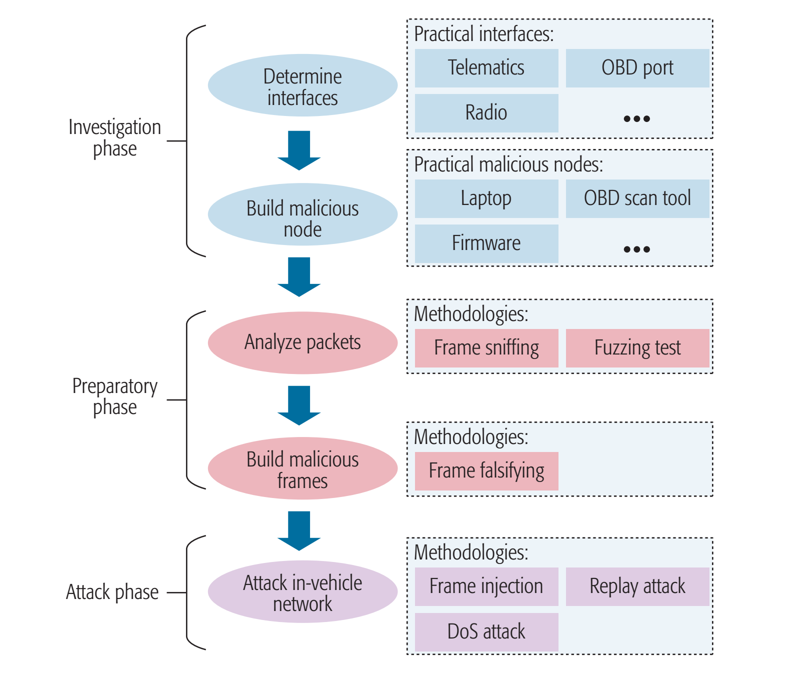

FIGURE 3. The general procedure is divided into three phases. In the investigation phase, based on the chosen interface the malicious node which can be used to sniff the traffic on the CAN bus needs to be built. In the preparatory phase, according to the analysis of historically recorded frames, adversaries build the fake frames containing falsified data. During the attack phase, adversaries are able to implement kinds of attacks by injecting fake frames.

A General Attack Procedure

Figure 3 shows a general attack procedure that can be used to attack an in-vehicle network. The procedure is divided into three phases: investigation phase, preparatory phase and attack phase. In the investigation phase, the interface that can be used to access the target in-vehicle network needs to be determined, such as the OBD port and telematics system. Based on the chosen interface, the malicious node that operates frame sniffing and frame injection can be built, such as a laptop, a reprogramming ECU or a telematics system infected by malware. In the preparatory phase, the malicious node sniffs the frames transmitted on the CAN bus. Since the CAN frames are broadcasted to all the nodes deployed on the CAN bus, all kinds of CAN frames can be gotten. Further analysis can be completed by analyzing the historically recorded frames and fuzzing test. Fake frames are built by falsifying the frame data. In the attack phase, adversaries are able to implement a replay attack and DoS attack by injecting corresponding frames. It should be noted that the fake frames built in the preparatory phase cannot be used to attack the cars of different models. Attackers have to repeat this attack procedure if they want to implement attacks on a car of a new brand.

Experimental Studies

In the last decade, there have been many studies on attacking in-vehicle networks. These experimental works are significant to the future studies.

Simulating Attacks: In [3], Hoppe and Dittman

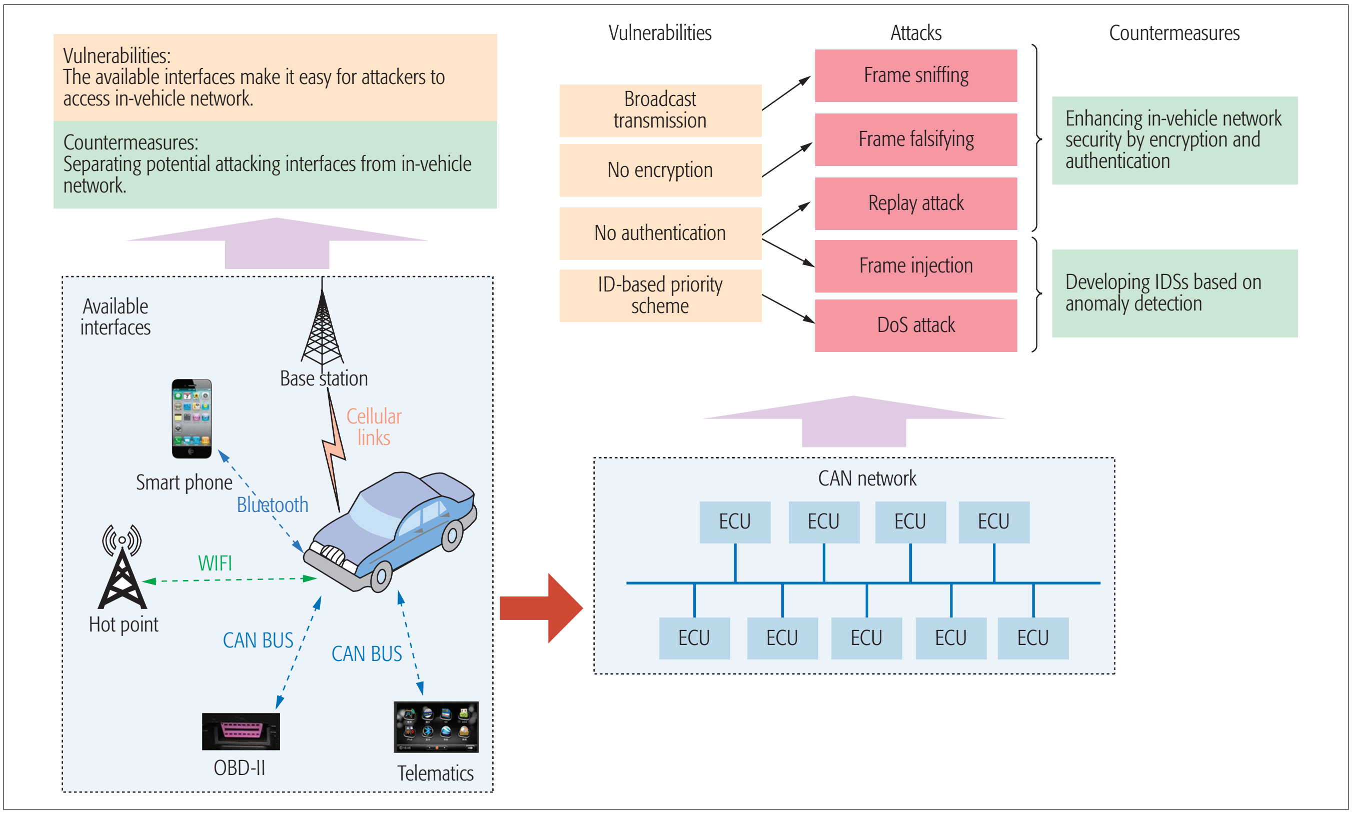

FIGURE 4. In-vehicle network has several intrinsic vulnerabilities, such as broadcast transmission, no authentication, no encryption, ID-based priority scheme and available interfaces, which makes in-vehicle network vulnerable to malicious attacks. Owing to these vulnerabilities, adversaries can easily implement various attacks on in-vehicle network, including frame sniffing, frame falsifying, replay attack, frame injection, and DoS attack. Corresponding countermeasures can be used to protect in-vehicle network from these kinds of attacks, such as encryption and authentication, developing IDS, and separating potential attack interfaces from in-vehicle network.}

implemented a frame sniffing and replay attack in a simulated environment of a simplified in-vehicle network. A software named CANoe was used to build the simulating environment consisting of a high-speed CAN bus, a low-speed CAN bus and several ECUs. These two buses are bridged by a gateway, and connect all the ECUs. Two scenarios were considered: a pure replay attack and frame sniffing. In the first scenario, the authors recorded the frames transmitted on the CAN bus while opening the window, then they took control of the window by sending the recorded frames to the CAN bus. In the second scenario, they controlled a malicious ECU that is deployed on the low-speed CAN bus to obtain the information transmitted from the high-speed CAN bus. This study shows the vulnerabilities of a CAN network. Frame sniffing and replay attacks can be easily implemented on the CAN bus.

In [4], Hoppe *et al.* used automotive components as the experimental environment. The authors utilized a simple replay attack to stop three components from working, i.e., the window lift, warning lights and airbag system. The approaches they used are described as follows. Once the valid frames were transmitted on the CAN bus, the authors controlled the malicious node to send frames containing the opposite command immediately. In this way, these three components could not work since the CAN bus was seriously interfered.

Practical Attacks: Koscher *et al.* are the first group to attack a real car [5]. The authors succeeded in controlling a wide range of modules in the target car by injecting corresponding frames to the CAN bus. In this study, the OBD port was chosen as the interface to access the in-vehicle network. A self-developed software named CARSHARK was used to operate frame sniffing and frame injection. They observed the traffic on the CAN bus and recorded plenty of valid frames. The recorded frames were analyzed by a fuzzing test (i.e., researchers test random frames iteratively). The frames directing at specific functions were obtained. Based on the analysis of the CAN frames, they were able to take control of the target modules by commanding the malicious node (i.e., the laptop with CARSHARK) to inject corresponding frames to the CAN bus. The modules controlled by the researchers are summarized as follows: radio, instrument panel cluster, body controller, engine, brakes, and heating ventilation air conditioning (HVAC) system. In addition, they easily disabled the communication of each of the components on the CAN bus by implementing a DoS attack.

Checkoway *et al.* in [6] made an analysis of the attack interfaces. The authors divided the interfaces into three types: indirect physical access (including the OBD port, USB and CD player), short-range wireless access (including Blutooth, WIFI and RFID) and long-range access (including FM, satellite radio and remote telematics systems). The vulnerability of each type of interface was described in details.

Wireless Attacks: In [7], Woo *et al.* implemented a wireless attack in a real car environment. They attacked the target in-vehicle network with the help of a smart phone that was paired with an OBD scan tool by Bluetooth. The authors took control of an ECU by commanding the malicious app installed on the smart phone. The attack was divided into two phases: preliminary phase and actual attack phase. During the preliminary phase, many valid CAN frames were gotten by a laptop that was connected to an additional CAN port. A malicious app that can operate frame injection was developed. During the actual attack phase, the malicious app sent appropriate frames to the CAN bus via the OBD port. Four types of attacks were implemented: distortion of the dash board, engine stop, handle control and acceleration.

Countermeasures for In-Vehicle Networks

The intrinsic vulnerabilities of the CAN bus and the available interfaces make in-vehicle networks vulnerable. Corresponding countermeasures are proposed to enhance in-vehicle network security. In addition, the vulnerabilities of in-vehicle networks, the attacks adopted by researchers and corresponding countermeasures are summarized in Fig. 4 to make the article clearer.

Enhancing In-Vehicle Network Security by Encryption and Authentication

The characteristics of no encryption and no authentication, as discussed earlier, make a CAN network vulnerable to adversarial attacks. Hence, enhancing in-vehicle network security by the functions of encryption and authentication is one of the effective countermeasures to provide confidentiality and reliability for a CAN frame. Four issues need to be carefully considered. First, encryption and authentication require efficient key management, including key distribution and key update. Second, the addition of message authentication code (MAC) to CAN frames definitely degrades the transmission efficiency. Third, providing encryption and authentication for CAN frames results in unexpected delay, which influences the vehicle maneuverability, especially for time-critical components. Fourth, because of a great number of products that are suited for a traditional CAN bus, the compatibility between existing devices and the CAN bus enhanced by encryption and authentication should be well addressed.

In [7], Woo *et al.* proposed a security protocol according to CAN specifications. AES encryption is employed to encrypt the CAN frames. A 32-bit MAC, which is composed of the first 16 bits of the extended ID field (i.e., the CAN 2.0B standard) and the 16-bit CRC field, is used to verify the identity of the transmitted ECU. Using a counter to record the generation of frames makes the proposed protocol robust to replay attack. In [9], Nilsson et al. proposed a mechanism to transmit a MAC by different frames. A 64-bit MAC is divided into four 16-bit parts and transmitted in four frames. This mechanism can improve the transmission efficiency of the authenticated frames.

The characteristics of no encryption and no authentication make a CAN network vulnerable to attacks. Hence, enhancing in-vehicle network security by the functions of encryption and authentication is one of the effective countermeasures to provide confidentiality and reliability for a CAN frame.

Developing IDSs Based on Anomaly Detection

The limited computing power, memory, and communication capacity of the electronic devices on a vehicle are the main limitations on applying conventional defense mechanisms to protect in-vehicle networks. The common intrusion detection systems (IDSs) designed for CAN networks usually follow mis-use or anomaly based attack detection methods [10]. Mis-use based detection uses the characteristics of already known attacks to identify specific attacks, e.g., DoS attacks, while the unknown attacks cannot be identified by mis-use based detection. Anomaly based detection is aimed at discovering abnormal frames transmitted on the CAN bus, which can identify the unknown attacks. Considering that the range of valid CAN frames is rather small and the possible attacks are varied, it is better to use an anomaly based detection technique in the IDS to detect attacks on in-vehicle networks.

Three issues need to be considered while designing an anomaly based IDS. First, the detectors that are used to detect abnormal frames should be appropriately deployed. ECU-based detection, i.e., placing a detector on each ECU, can easily get the source of abnormal frames. The deployment of detectors will significantly increase in-vehicle network complexity. Network-based detection, i.e., placing detectors on an in-vehicle network, can detect the abnormal frames transmitted on the CAN bus. While in this mode, the source of abnormal frames is unknown. Second, the detecting methodologies designed for IDS are of great importance. This issue has attracted research attentions for many years. Larson *et al*, [11] developed security specifications for ECUs, as the predefined patterns to detect abnormal frames. The authors compared CAN frames with predefined patterns to verify whether the frames are valid or not. Song *et al*. [12] pointed out that the valid frames with different CAN IDs have unique time intervals. According to this characteristic, they developed an IDS based on the analysis of the time intervals of CAN frames.

In [13], Kang *et al*. proposed an intrusion detection technique using a deep neural network (DNN). The authors trained the DNN via a great number of valid CAN frames and tested the CAN frames with the trained DNN. In the experiment, 3900 frames were tested in only 7 to 8 milliseconds, and 99.8 percent of the hacking frames were detected when the DNN contained five layers. Third, the accuracy of IDSs should be carefully considered. In [14], the researchers proposed a dynamic anomaly detection scheme for MANET. With this scheme, if the tested state is judged as normal, the corresponding data set will be used as the training data set. Compared with the method of using only the initial training data set, the average false positive rate of dynamic anomaly detection is decreased by more than 10 percent. A similar dynamic anomaly detection scheme for CAN networks is also worth considering.

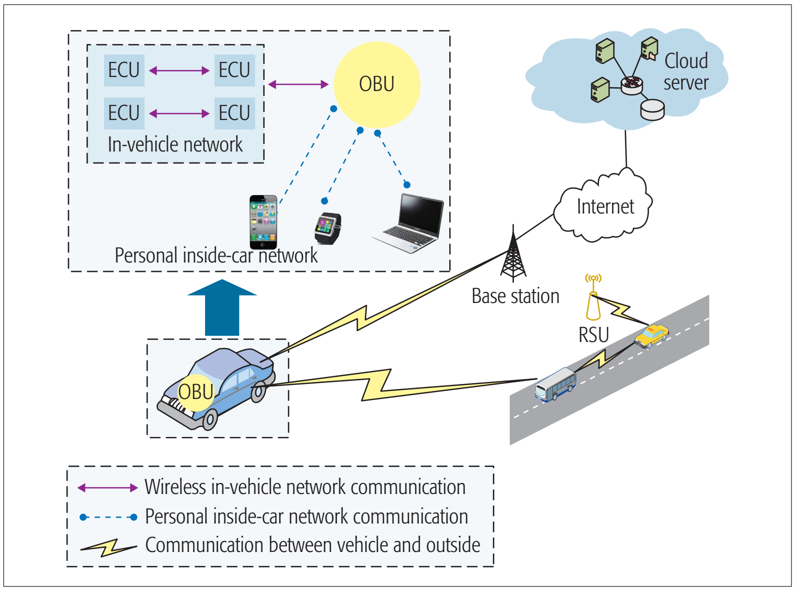
ECU
ECU
ECU
ECU
In-vehicle network
OBU
Personal inside-car network
Cloud server
Internet
Base station
RSU
OBU
Wireless in-vehicle network communication
Personal inside-car network communication
Communication between vehicle and outside

FIGURE 5. In the enhanced vehicular network architecture, ECUs use dedicated spectrum to communicate with each other. Personal inside-car network composes of mobile devices, wearable devices, OBU and in-vehicle network. On board unit (OBU) is employed to communicate with other entities such as road side units (RSUs), cellular base stations and vehicles.

Separating Potential Attacking Interfaces from In-Vehicle Networks

Adversaries can easily access an in-vehicle network and implement attacks via the interfaces of the in-vehicle network, so separating interfaces from an in-vehicle network can definitely enhance the in-vehicle network security. The common interfaces are the OBD port and telematics system. Since the OBD port is designed to diagnose the state of a car, it can hardly be separated from the in-vehicle network. One of the solutions is to add detector to OBD port to detect abnormal frame injection. The telematics system, which provides entertainment, Global Position System (GPS) and radio for drivers, can be regarded as an independent part in a car. The separation of the telematics system and the in-vehicle network can decrease the risk of suffering adversarial attacks.

Challenges and Future Directions

In this section, we present an enhanced vehicular network architecture based on the on-board unit (OBU), which is shown in Fig. 5, to simplify the in-vehicle network and to provide connectivity between the in-vehicle network and the outside hybrid network. The enhanced vehicular network architecture consists of a wireless in-vehicle network, personal inside-car network and outside hybrid network. In the wireless in-vehicle network, ECUs interact with each other to control the vehicle, such as braking and acceleration. The personal inside-car network is composed of an in-vehicle network, wearable devices, mobile devices and OBU. The OBU is employed to communicate with the outside hybrid network.

Wireless In-Vehicle Networks

The conventional CAN bus has limited communication capacity. The first constraint is the limited bandwidth provided by the CAN bus. The rate of the high-speed CAN bus is about 500 kb/s, which cannot meet the demand of the expected large amounts of sensor data. Due to the broadcast and half-duplex characteristics of the CAN bus, the ECUs cannot transmit frames simultaneously. The second constraint is the limited space for cables. With the increasing intelligence of modern vehicles, a growing number of sensors and ECUs will be deployed on the CAN bus to detect road conditions, vehicle state and driver's health. The wired CAN bus connects up the ECUs by cables, which increases the weight of the vehicle. In light of these constraints, a wireless in-vehicle network is proposed as the next frontier of in-vehicle networks.

Wireless in-vehicle networks can simplify the structure of modern vehicles and reduce the cost of cables. Several potential issues with wireless in-vehicle networks are non-negligible. First, wireless in-vehicle networks need a suitable protocol and dedicated spectrum, which can significantly increase the quality of service (QoS) of wireless in-vehicle network communication. The available short-range wireless technologies, such as Bluetooth, RFID and ZigBee, all operate in the industrial, scientific and medical (ISM) frequency band. If in-vehicle networks adopt these wireless communication technologies, it will be seriously interfered by other devices, such as smart phones, laptops and wearable devices. Second, the interference generated by other vehicles should be evaluated. The accumulated interference from other vehicles may result in the outage of wireless in-vehicle network communication.

Personal Inside-Car Networks

The personal inside-car network, composed of mobile devices, wearable devices, the OBU and in-vehicle network, will bring comfort and safety to drivers. According to the identity indicated by mobile devices and the driver state provided by wearable devices, an in-vehicle network is able to take corresponding actions, such as adjusting seat position and regulating temperature, to provide a comfortable driving environment for the driver. In addition, the in-vehicle network messages created by the ECUs can be sent to a cloud server to implement real-time vehicle diagnosis. Even more, according to the vehicle diagnosis results, vehicles that have car troubles can be marked as "dangerous vehicles" in the VANET, warning other drivers to prevent accidents.

Several potential issues with personal inside-car networks should be taken into consideration. First, the recognition of the driver's devices, including mobile devices and wearable devices, should be accurate. Recognizing the driver's devices is the foundation of personal inside-car network functions. Second, the interference generated by other devices may seriously influence the connectivity of a personal inside-car network, especially in urban areas with a high density of vehicles and mobile devices. Third, the "dirty drivers" will significantly deteriorate road conditions. In order to get more space on a road crowded with traffic, the "dirty drivers" may operate abnormal behaviors on purpose or control their OBUs to send "dangerous vehicles" statements to the VANET.

Outside Hybrid Networks

Nowadays, vehicles are almost isolated from external networks. The connectivity provided by the telematics system is limited, and cannot meet the increasing data demand of vehicles. In the enhanced vehicular network architecture, vehicles are designed to connect to hybrid networks, including VANETs, cellular networks and the cloud. OBUs are employed to communicate with other entities such as road side units (RSUs), cellular base stations and vehicles. In the enhanced vehicular network architecture, a cloud server is responsible for providing computing resources to the vehicles, such as diagnosing the vehicle according to the in-vehicle network messages. A VANET provides communications among the vehicles on the road, such as broadcasting the “dangerous vehicles” to the automobiles in the VANET. Vehicles are regarded as the mobile nodes that can access the Internet. OBUs are employed by cars to get the messages transmitted from the Internet and represent them to drivers.

Several issues with outside hybrid networks should be well addressed. At first, the approaches used by cloud servers to analyze the messages generated by different in-vehicle networks need to be found out. Because of the wide range of vehicle brands, the in-vehicle messages of various vehicles may be totally different. Although the standardization of in-vehicle networks is an efficient solution to this problem, short-term countermeasures are also necessary. Then, integrating the cloud services of different automobile manufacturers and providing a standard for vehicle cloud services are of great importance. Beyond that, the confidentiality of the transmitted messages, especially for the data created by wearable devices and mobile devices, is a potential issue deserving careful consideration.

Because of the wide range of vehicle brands, the in-vehicle messages of various vehicles may be totally different. Although the standardization of in-vehicle networks is an efficient solution to this problem, short-term countermeasures are also necessary.

Spectrum for VANET Communication (Table 2)

Diverse applications in enhanced vehicular networks present increasingly high requirements on VANET communication technologies. IEEE 802.11p based Dedicated Short Range Communication (DSRC) and Long-Term Evolution (LTE) based LTE-V are two potential candidates for VANET communication (Table 2).

IEEE 802.11p is an enhancement of the IEEE 802.11 standard, which adds Wireless Access in Vehicular Environments (WAVE). Although DSRC is considered to be the first standard designed for V2X communication, it is still in the field-trial stage. In the United States, 75 MHz of bandwidth (5.850–5.925 GHz) has been allocated for DSRC communication. In Europe, 50 MHz has been provided for vehicular communication at 5.855–5.905 GHz. In particular, 30 MHz at 5.875–5.905 is solely intended for road traffic safety applications, while a 20-MHz band at 5.855–5.875 GHz is assigned for nonsafety-related applications. In Japan, the spectrum allocation for DSRC is not in alignment with the allocations in Europe or the USA. The 5.8 GHz band (5.770–5.850 GHz) has been allocated for V2I communication. The 700 MHz band (715–725 MHz) has been allocated for V2V communication. LTE-V is a systemic V2X solution based on time-division LTE (TD-LTE). LTE-V includes two modes: LTE-V-cell and LTE-V-direct. LTE-V-cell is proposed to support V2I communication, while LTE-V-direct supports V2V communications. In 2016, the government of China indicated that 5.905 - 5.925 GHz was allocated for the technical validation of LTE-V-direct.

Regions	Applied technologies	Bandwidth
USA	DSRC	75 MHz (5.850–5.925 GHz)
Europe	DSRC	50 MHz (5.855–5.905 GHz)
Japan	DSRC	80 MHz (5.770–5.850 GHz) 10 MHz (715–725 MHz)
China	LTE-V	20 MHz (5.905–5.925 GHz)

TABLE 2. Spectrum for VANET communication in different regions.

Potential Adversarial Threats for Vehicles

The adversarial threats for vehicles in the enhanced vehicular network architecture are different from that for the conventional vehicles. The increasingly rich communications between vehicles and the outside hybrid network make vehicles easily vulnerable to cyber attacks. The enhanced network architecture is based on OBUs, which means it is easy for adversaries to steal all the data generated from the vehicle by sniffing the data flow through the OBU. In addition, because of the wireless communications among ECUs, the adversaries are able to snoop in-vehicle messages directly rather than by the interfaces.

Conclusions

In this article, we reviewed most of the experimental works and pointed out future challenges for in-vehicle networks. Due to the intrinsic vulnerabilities of in-vehicle networks and the increasingly rich interfaces to provide connectivity between the in-vehicle network and outside networks, adversaries can easily implement attacks on in-vehicle networks, especially for the emerging wireless in-vehicle networks.

References

[1] N. Lu *et al.*, “Connected Vehicles: Solutions and Challenges,” *IEEE Internet of Things J.*, vol. 1, no. 4, 2014, pp. 289–99.

[2] M. Wolf, A. Weimerskirch, and T. Wollinger, “State of the Art: Embedding Security in Vehicles,” *EURASIP J. Embedded Systems*, vol. 2007, no. 2, 2007.

[3] T. Hoppe and J. Dittmann, “Sniffing/Replay Attacks on CAN Buses: A Simulated Attack on the Electric Window Lift Classified using an Adapted CERT Taxonomy,” *Proc. 2nd Workshop on Embedded Systems Security*, 2007.

[4] T. Hoppe, S. Kiltz, and J. Dittmann, “Security Threats to Automotive CAN Networks — Practical Examples and Selected Short-Term Countermeasures,” *Proc. SAFECOMP*, 2008.

[5] K. Koscher *et al.*, “Experimental Security Analysis of a Modern Automobile,” *Proc. IEEE Symposium on Security and Privacy*, 2010.

[6] S. Checkoway *et al.*, “Comprehensive Experimental Analyses of Automotive Attack Surfaces,” *USENIX Security*, 2011.

[7] S. Woo, H. J. Jo, and D. H. Lee, “A Practical Wireless Attack on the Connected Car and Security Protocol for In-Vehicle CAN,” *IEEE Trans. Intelligent Transportation Systems*, vol. 16, no. 2, 2015, pp. 993–1006.

[8] W. Zeng, M. A. S. Khalid, and S. Chowdhury, “In-Vehicle Networks Outlook: Achievements and Challenges,” *IEEE Commun. Surveys Tutorials*, vol. 18, no. 3, 2016, pp. 1552–71.

[9] D. K. Nilsson, U. E. Larson, and E. Jonsson, "Efficient In-Vehicle Delayed Data Authentication Based on Compound Message Authentication Codes," *Proc. Vehicular Technology Conf.*, 2008.

[10] Z. M. Fadlullah *et al.*, "Intrusion Detection System (IDS) for Combating Attacks Against Cognitive Radio Networks," *IEEE Network*, vol. 27, no. 3, 2013, pp. 51–56.

[11] U. E. Larson, D. K. Nilsson, and E. Jonsson, "An Approach to Specification-based Attack Detection for In-Vehicle Networks," *Proc. IEEE Intelligent Vehicles Symposium*, 2008.

[12] H. M. Song, H. R. Kim, and H. K. Kim, "Intrusion Detection System Based on the Analysis of Time Intervals of CAN Messages for In-Vehicle Network," *Proc. Int'l. Conf. Information Networking (ICOIN)*, 2016.

[13] M.-J. Kang and J.-W. Kang, "A Novel Intrusion Detection Method Using Deep Neural Network for In-Vehicle Network Security," *Proc. IEEE 83rd Vehicular Technology Conf.* (VTC Spring), 2016.

[14] H. Nakayama *et al.*, "A Dynamic Anomaly Detection Scheme for AODV-Based Mobile Ad Hoc Networks," *IEEE Trans. Vehic. Technol.*, vol. 58, no. 5, 2009, pp. 2471–81.

Biographies

Jiajia Liu [S'11, M'12, SM'15] is a full professor at the School of Cyber Engineering, Xidian University. His research interests cover wireless mobile communications, FiWi, IoT, and so on. He has published more than 50 peer-reviewed papers in many prestigious IEEE journals and conferences, and currently serves as an associate editor for *IEEE Transactions on Computers*, and *IEEE Transactions on Vehicular Technology*, an editor for *IEEE Network*, a guest editor of *IEEE TETC* and *IEEE IoT Journal*. He is a Distinguished Lecturer of the IEEE Communications Society.

Shubin Zhang received his B.S. degree in computer science from Xidian University in 2014. He is currently working toward the Ph.D. degree at the School of Cyber Engineering, Xidian University. His research interests include vehicular network security and controller area network security.

Wen Sun [S'11, M'16] received her Ph.D. degree in electrical and computer engineering from National University of Singapore in 2014. She is currently an associate professor with the School of Cyber Engineering, Xidian University. Her research interests cover a wide range of areas including body sensor networks, IoT, participatory sensing, and 5G.

Yongpeng Shi received his B.S. degree in electronic information science from Shaanxi Normal University in 2001 and an M.S. degree in computer science from Xidian University in 2008. He is pursuing his Ph.D. degree at Xidian University. His research interests include network function virtualization and cloud computing.