Cyber Threats to Automotive Technology

Luis Emilio Núñez Morales

Lúsofona University of Porto, Portugal
University of Vigo, Spain

emilio-0111@hotmail.com

Abstract. In the last decades, automobiles have been transformed from mechanical machines, with barely any electrical device, to advanced machines that contain hundreds of electrical components. All of this new features have to be managed and controlled by software, turning today's vehicles into technological devices with up to millions of lines of code. This amount of software opens up the opportunity for hackers to find vulnerabilities and exploit them, being able to cause a lot of damage. This paper will discuss the importance of cyber security in the automotive industry and how it is becoming a new dimension of quality in vehicles. We will talk about cyber attacks on cars, focusing on remote attacks, and overview some of the methods and standards used to prevent those cyber attacks. Finally, this paper will show two case studies of attacks on cars, one well known from 2015 (Jeep Cherokee attack) and another one from a more recent year, with the purpose of arriving to a conclusion on whether or not there has been any improvement.

Keywords: Cyber Security, Automotive Industry, Cyber Threats, Case Study, Standards, Remote attacks.

1. Introduction

In recent years, software development has had a more or less relevant weight for the industrial sector, but increasingly, the integration of software in the production chains of organizations makes it an essential element in the process of digital transformation of the production chains. Manufacturers are integrating new technologies, such as Internet of Things (IoT), cloud computing and analytics, and AI and machine learning into their production facilities and throughout their operations. This implies the appearance of a Fourth Industrial Revolution.

The Fourth Industrial Revolution, also known as Industry 4.0, is changing the way business operate and therefore the environments in which they are forced to compete. Industry 4.0 can improve business operations and revenue growth, transforming products, supply chain, and customer expectations [1].

Nevertheless, with the advantages of applying software in the industry, the threats of its use appear. The vulnerabilities of the different industries are increased by the cyber threats that come with the application of the software in the production operations and in its use in the final products.

The automotive industry, evolving alongside Industry 4.0, is directly affected by its pros and cons. The exponential growth in the amount of software found in automobiles today is a great challenge for manufacturers, who have to respond to a constantly evolving market for software innovations while addressing the threats of adding large amounts of software in their final products.

This paper will focus on the threats posed by adding software functionalities to the final product of automotive industries, their consequences for manufacturers and customers, and ways to mitigate them. This paper will also address the current regulations in Europe regarding cyber security in automobiles, which will determine a large part of the production of automobiles. Finally, we will attend to two case studies, separated in time, and analyze the evolution of both the functionalities and the threats of software in vehicles.

2. Cyber security as a quality feature for automobiles.

"Once, software was a part of the car. Now, software determines the value of a car," notes Manfred Broy, emeritus professor of informatics at Technical University, Munich and a leading expert on software in automobiles. "The success of a car depends on its software much more than the mechanical side." Nearly all vehicle innovations by auto manufacturers, or original equipment manufacturers (OEMs) as they are called by industry insiders, are now tied to software, he says [2].

In the last decades, automobiles have been increasing their functions, gadgets that increase safety, improve efficiency and contribute to the well-being and entertainment of the user. In order to do this, they make use of several ECUs, which contain the necessary software to monitor and control these functions.

With the increase in the number of functions in automobiles and their complexity, the number of ECUs required to control these functions has also increased. A decade ago, a luxury car could have up to 100 ECUs controlling the mechanical elements and hardware of the car, executing more than 100 million lines of code. With functions such as Cruise Control, Rear View Camera, Emergency Braking Systems or Parking Sensors becoming a standard in modern cars, today, we can find this amount of ECUs in more basic and lower-end cars. On the other hand, modern high-end cars like the Mercedes S-Class can have up to 150 ECUs as they have a large number of complex functions. This amount of code creates ample opportunity for cyber attacks, not only on the car itself but also on all components of its ecosystem (e.g., back end, infrastructure).

Cyber security in cars is becoming a new dimension to measure their quality [3] as well as the prestige of the manufacturer. OEMs will have to invest the necessary resources to face this new challenge in the automotive industry, protecting against cyber threats will be a difficult but necessary task in the coming years. The ability of

OEMs to provide online software updates on their cars will be one of the key pillars to ensure proper vehicle safety management.

3. Security threats: common attacks and countermeasures.

Currently, cars have gone from being mechanical machines with a minimum amount of electronic components to regulate their operation, to being machines with advanced technology that makes them smarter, more efficient and safer. Most manufacturers offer software services in their products, such as the possibility of interacting with a vehicle remotely via the Internet or knowing its GPS location from a smart phone. These services present vulnerabilities that can be exploited by malicious hackers, which can lead to serious accidents, data theft and damage the image of the manufacturer.

3.1. Remote attacks.

One of the most dangerous and powerful threats to modern automobiles is performing a remote attack. The nature of this type of attack allows the malicious hacker to get access to the target car, without the need of having any kind of physical contact with it. Hacking a car remotely can help the intruder access into the car's internal network without being noticed by the owner of the car. Remote safety attacks against automobiles are generally divided in three stages [4].

The first stage is to gain access to the vehicle internal network. An attacker can gain access to an automobiles network by using ECUs that connect to the car with its surroundings. One way of doing so could be sending a wireless signal to a listening ECU on the car, subsequently injecting code. This will allow the attacker to send malicious messages into the car's networks, controlling the desired ECU. There are a bunch of point entries in modern cars. These include Bluetooth, Remote Keyless Entry, RFIDs, Tire Pressure Monitoring Systems, WiFi, and Dedicated Short-Range Communications among others.

Gaining access to these ECUs, whose job is only receive and process radio signals, will not give attackers the opportunity to perform a cyber physical attack -attack that result in physical control of various aspects of the automobile- . In order to perform a cyber physical attack, it will be necessary to inject messages onto the internal automotive network in an attempt to communicate with safety critical ECUs, such as those responsible for steering, braking, and acceleration. This is the second stage of a remote attack. The attacker will somehow have to get messages from the bridged network to the internal network where the target ECU lives. One way of doing so is to escalate privileges inside the CAN network (the network inside a modern car where control commands for various components inside the vehicle travel), creating a bridge between the firstly infected ECU, the one that interacts with the external world, and the target ECU, that controls critical features of the vehicle.

Once the ECU is been wirelessly compromised, and after being able to use it to communicate to the ECUs that control safety features, the attacker will want to send them instructions for malicious purpose. This is the final stage, trying to make the

target ECU behave in some way that compromises vehicle safety. In order to perform such action, the attacker will have to reverse engineer the messages on the inner network and figure out the format of the instructions, that way the attacker can replicate those instructions and execute them at will, resulting on physical actions such as braking or steering the wheel. Since each manufacturer (and perhaps each model and even each year) use different data in the messages on the bus, the message reverse engineering process requires a large amount of work and will be manufacturer specific.

3.1. Countermeasures.

In order to prevent these malicious attacks, automobiles must be built with cyber security in mind. In this paper we are going to overview security practices and standards that will help avoid not only remote attacks on vehicles, but every kind of cyber security threats.

Automotive threat modeling. Threat modeling involves understanding the complexity of the system and identifying all possible threats to the system. During the formation of security requirements, these threats are analyzed based on their criticality and likelihood, and a decision is made whether to mitigate the threat or accept the risk associated with it [5].

When it comes to automotive threat modeling, there is one particular framework popular in the computer industry, that seems to be the most suitable for the automotive industry, STRIDE. This framework covers the main six board categories of threats: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service and Elevation of Privilege. This threats are always present in remote attacks on automobiles, they are the ones that make this type of attacks happen, so seems to be appropriate to adopt a framework that focuses on covering these threats.

AUTOSAR. AUTomotive Open System ARchitecture is a global partnership of leading companies in the automotive and the software industry to develop and establish the standardized software framework and open E/E system architecture for intelligent mobility [6].

The standard comprises a set of specifications describing software architecture components and defining their interfaces. The principal aim of the standard is to master the growing complexity of automotive electronic architectures.

The AUTOSAR standard defines security mechanisms that can be used by the software modules implemented into the vehicle system. It further specifies interfaces and procedures to provide secure on-board communication, and the exact implementation is left for the OEMs to decide on. OEMs choose the cryptographic algorithms and encryption techniques which they want to implement and use in the vehicle system.

4. Cyber security standards and regulations in Europe.

In 1885, Karl Benz invented a car that is credited as the first car in the world to be powered by fuel.The design of the car was based on a horse carriage—the difference was that instead of a horse, he put an engine [7]. At that time, horses were considered to help avoid collisions and humans were not going to be able to carry out this task. Those who thought this way were not going wrong, the advances that vehicles brought with them generated new dangers. It was common to drive at high speed or under the influence of alcohol, so it was necessary to create a series of rules and make them known to society to have control over the behavior of drivers. Thus, Road Safety was born.

The industry has a relevant role in this whole process, since ensuring that the cars were safe enough for their occupants was essential for Road Safety to continue advancing. In 1930 the first cars with hydraulic brakes and steel frames came onto the market, and in 1959 Volvo began to install one of the most important passive safety features: the three-point seat belt.

Nowadays, as we have cover already on this paper, vehicles have been incorporating software functionalities and becoming data centers on wheels. With the appearance of new functionalities in automobiles, new rules appear so that they continue to be safe. The new regulations address the risks which arise from the increasing digitalization of vehicle functions and the connection of vehicles with their environment (connectivity). The regulations' objective is to generate a harmonized regulatory framework for vehicle development to enable international vehicle trade.

On March 4, 2021, UNECE publishes regulations R155 and R156, for cyber security in automobiles and software updates respectively [8]. UNECE regulation 155 (Cyber Security) introduces a cyber security Management System (CSMS) in automotive on organization level. This regulation encourages the use of standards in the production of automobiles that address the cyber security of the product.

UNECE regulation 156 (SW Updates) introduces a legal framework for remote updates (over-the-air) with a Software Update Management System (SUMS). The main objective of this regulation is to make sure OEMs provide secure software updates with the guarantee that vehicle safety is not reduced.

4.1. ISO 26262.

Automotive manufacturers and suppliers must be certified, as they must offer formal third-party assurance according to the safety standard, such as ISO 26262. The ISO 26262 [9] standard defines a framework, an application model, the activities to be carried out, the methods to be used and the results, offering manufacturers a common mechanism to measure and document the safety of an automotive system. It is necessary to manage functional safety and regulate the development of automobiles at the hardware, software and system level throughout their life cycle.

5. Jeep case study.

This section of the paper will overview the case study of the Jeep Cherokee cyber security attack that was performed on 2015 by two white hat hackers, Charlie Miller and Chris Valasek. The mentioned hackers elaborated a document [10] with the details of the attack that allowed them to take control of the Jeep Cherokee. In this document we will try to analyze the entire process of said attack without going into too specific details, and we will point out those security threats that should have been covered at the time of production of this vehicle.

5.1. The entire exploit chain.

Identify target: There are a variety of entry points on this vehicle, such as Bluetooth, WiFi or radio. The most interesting one is the cellular connection functionality, since it seems to be the most powerful one as it has the larger range. If you knew the Vehicle Identification Number (VIN) or GPS, you could scan the IP ranges where vehicles are known to reside until you found one with corresponding VIN or GPS. We could target one car only or use a worm to hack a number of them.

Get an SSH connection to be able to run code in the car´s system: Once we have the IP address of the vehicle we can port scan the default gateway and examine if there are any ports open. At the time the hackers were performing the attack, the 2014 Jeep Cherokee had several ports open. Here we find the first vulnerability of this vehicle. This vulnerability presents a big threat, as it provides a way for attackers to get into the car´s internal network. The port that is most interesting for us is the 6667, that was connected to a D-Bus. D-Bus (Desktop Bus) is an inter-process communication system (IPC) and a remote procedure call (RPC), for software applications in order to communicate with each other. This D-Bus message daemon is part of an infotainment, Wi-Fi connectivity, navigation, apps and cellular communications system called Uconnect 8.4AN/RA4 radio manufactured by Harman Kardon. D-Bus can require authentication. On the Jeep head unit, the authentication is open to anonymous action. Here is another security breach. The D-Bus should not offer the possibility of logging into the system with an anonymous authentication.
At this point we can interact with the D-Bus services on the Jeep. Some of them will allow us to acquire direct interaction with the head unit, giving us the ability to adjust the volume of the radio, access PPS data, among other things.
There are other D-Bus services that actually provide an "execute" method which is designed to execute arbitrary shell commands. This is a big security flaw. There should be no way to be able to execute shell commands from outside of the vehicle's

interior system. This vulnerability should have been covered before the vehicle was put on the market. Taking advantage of this vulnerability we can establish a reverse shell to obtain an interactive shell session on the vehicle's OMAP (Open Multimedia Applications Platform) chip which manages the majority of the functionality of the Uconnect system.

Flash the v850 with modified firmware: In order to perform a cyber physical attack we will need to be able to send instructions through the CAN bus of the vehicle. Controller Area Network (CAN) is a protocol based on messages designed to allow vehicle's Electronic Control Units to interact with each other. The OMAP chip, on which we have code execution on after the D-Bus exploit, cannot send CAN messages. It can, however, communicate with the v850 chip which can send CAN messages. In order to use the v850 to send CAN messages we need to upload a modified firmware that will allow us to send those messages. Reverse engineering the update file for the v850 should allow us to flash the v850 with the modified firmware.

Perform cyber physical actions: Once you can send CAN messages via remote exploitation, it is simply a matter of figuring out which ones to send to affect physical systems. There are two types of CAN messages, normal and diagnostic. Normal messages are seen all the time on the bus during normal operation. Diagnostic messages typically are only seen when a mechanic is testing or working on an ECU, or some other unusual circumstance is occurring. Using normal CAN messages we can manipulate physical features such as the turn signals, the locks and the tachometer. Diagnostic messages are more powerful than normal messages, however most ECUs will ignore diagnostic messages if the car is traveling at speed, usually faster than 5-10 mph. Therefore, these attacks can typically only be performed when the car is traveling rather slowly, unless the attacker can figure out how to forge a speed used to determine if diagnostic messages should be accepted. Using diagnostic messages we could kill the engine, disable the brakes and even get control of the steering wheel.

6. Tesla case study.

In this section we will study two Tesla cyber security researches made by Keen Security Lab of Tencent, a team focused in cutting-edge security research of mainstream PC/Mobile operating systems, applications, cloud computing technologies, IOT smart devices, etc. One of the researches describes how they remotely compromised the ECUs of Tesla cars [11], and the other one focuses on the vulnerabilities of the Tesla autopilot system [12]. We will start with the 2018 document that shows how they managed to gain remote access and then we will

continue with the 2019 document where they show how they compromised the autopilot system.

6.1. Remotely obtain root privilege of APE (AutoPilot ECU).

In 2017, Tesla had built-in a Webkit based web browser in their vehicles to allow the users of the car to navigate through the internet. The Tesla Model S had this browser inside a 17-inch touchscreen Center Information Display (CID) in the middle of the dash. Keenlab Team found an Use After Free (UAF) vulnerability in Webkit. Basically, an UAF vulnerability occurs when a pointer to an object that has been freed is referenced, an attacker could modify an unintended memory location that potentially can lead to code execution. Keenlab Team managed to insert the right instructions in the right memory address and gained arbitrary code execution inside the CID as a result.

The AutoPilot ECU (APE) module in the Tesla has control of the systems that provide driving assistance to the driver, such as lane centering, adaptive cruise control or self-parking among others. Unlike CID, there are few interfaces on APE interacting with the outside world, making it difficult to hack into APE. Keenlab Team took advantage of a vulnerability in the update file of the APE that allowed them to modify the existing firmware with a customized code that granted them a way to execute commands with ROOT privilege.

6.2. Remotely control the steering system.

APE is responsible for managing the steering system and the electronic speed control of the car whenever the is on Automatic Parking Control (APC) or adaptive Cruise Control (ACC) modes. The necessary communication between ECUs to manage the steering is made throw a CAN-bus system.

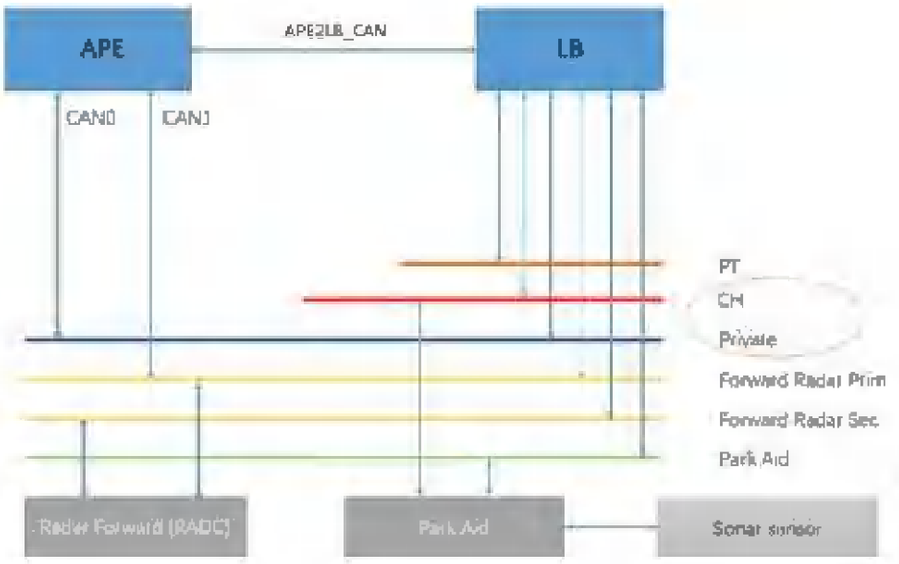

Fig. 1. CAN Bus System of APE.

After reverse engineering some CAN services, Keenlab Team managed to understand how the networking in the CAN bus system of APE works. They discovered that they needed to get access to the Power Train and Chassis CAN buses through the APE2LB_CAN and LB unit (see fig.1) in order to control the Electric Power Assisted Steering (EPAS) unit.

DasSteeringControlMessage (DSCM) is a CAN message produced by the Cantx service (a service associated with CAN-bus in APE) designed to control the steering system when the car is in ACC or APC modes. Keenlab Team managed to inject malicious code into the Cantx service and send DSCM messages that allowed them to control the steering of the car.

6.3. Attacks from physical adversary scenarios.

So far we have only covered remote attacks that took advantage of vulnerabilities in the software´s design of the vehicle (in both the Jeep and Tesla case study) to access the key components that manage important aspects of the car, allowing us to change the way they behave by sending malicious instructions. But now we are going to overview a different kind of security threat, one that does not require an intrusive attack to be exploited.

Auto wipers: Tesla´s auto wiper system uses a pure computer vision solution, based on a 120-degree fisheye camera and an artificial intelligence network to determine whether the wipers should be turned on. Keenlab Team found a way to deceive the

auto wiper system by displaying an image that used a noise function. The image displayed in front of the car's camera made the system think it was raining and therefore it turned on the wipers.

Lane Detection: The lane detection system also uses a computer vision solution, but includes more communication between various components of the car, as it has to know where the car is on the road and perform the necessary actions (such as steering the wheel) to keep the car in between the lines. There are two different types of lane recognition attack: eliminate lane attack and fake lane attack. The eliminate lane attack consists in blurring the road lane in the physical world, forcing the system to ignore that there is a lane on the road. This attack is quite difficult to perform because the Tesla's lane recognition system is designed to recognize even the most abnormal lanes (broken, occluded). But this feature has also it's cons, since it might recognize a lane where there isn't one. This is a fake lane attack. Keenlab Team managed to deceive the lane recognition system by just putting some small stickers on the road, making the vehicle think that there is a lane there. A fake lane attack could be extremely dangerous if the fake lane is pointing to the reverse lane of the road.

7. Conclusion.

In this paper we have talked about how software has been progressively introduced in the automotive industry. We pointed out the importance of building a strong and secure software architecture in automobiles, as it is becoming a new dimension to measure the quality of modern vehicles. We have focused on the structure of remote attacks and overviewed some countermeasures and good practices to avoid them. Furthermore, this document summarizes the objective of the new and necessary cyber security regulations in the automotive industry, which will help to target a better cyber security scene in the automobiles of the future.

In both the Jeep and the Tesla cases, we have seen security vulnerabilities that allowed intruders to get remote access to a ECU inside the vehicle that connects to the outside world. We have seen that it is possible to extend the attack using the vehicle's CAN bus to reach key ECUs that control very important aspects of the vehicle. In the Tesla case, it seemed to be a harder job to perform these attacks, which might let us think there has been improvement in cyber security in comparison with the Jeep case. But one thing that stands out for me in the Tesla case is how new vulnerabilities are coming together with new technology and features. It is undeniable that there has been a slight improvement in cyber security regarding known and past threats. Nevertheless, it seems that the desire to create new technologies and add new software in vehicles that can make them more advanced is overtaking the initiative to seek safer structures in these new technologies.

References

1. IBM: What is Industry 4.0? URL: https://www.ibm.com/topics/industry-4-0
2. Robert N. Charette: How Software is Eating the Car. IEEE Spectrum. (2021).
3. Ondrej B., Johannes D., Benjamin K, Klaus P, Gundbert, S.: Cybersecurity in automotive (2020)
4. Charlie Miller, Chris Valasek: A Survey of Remote Automotive Attack Surfaces. (2014)
5. Zhendong Ma, Christoph Schmittner: Threat Modeling for Automotive Security Analysis. (2016).
6. AUTOSAR: General Information About AUTOSAR. URL: https://www.autosar.org/about/
7. CarAdvise: What Was The First Car In The World (2019)
8. Henning Schweder: UNECE Vehicle Regulation for Cyber Security & Software Updates.
9. ISO.org: ISO 26262-1:2018 Road Vehicles - Functional Safety. (2018)
10. Keen Security Lab of Tencent: Over-The-Air: How We Remotely Compromised The Gateway, BCM, and Autopilot ECUs of Tesla Cars. (2018).
11. Keen Security Lab of Tencent: Experimental Security Research of Tesla Autopilot. (2019).
12. Charlie Miller, Chris Valasek: Remote Exploitation of an Unaltered Passenger Vehicle. (2015).