UNIVERSITÀ DEGLI STUDI DI CATANIA

Dipartimento di Matematica e Informatica

Dottorato di ricerca in informatica (Internazionale) XXXV Ciclo

Pietro Biondi

Automotive 2.0:

Security, privacy and safety in today's automotive domain

Ph.D. Thesis

Supervisor: Prof. Giampaolo Bella

Academic year 2021 - 2022

Declaration of Authorship

I, Pietro Biondi, declare that this thesis titled, "Automotive 2.0:
Security, privacy and safety in today's automotive domain" and the work presented in it is my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Abstract

The automotive security industry is expanding precisely because cars tend to be increasingly connected to each other. For this reason modern cars can be considered as a computer on wheels. Modern cars have many different types of electronic control units (ECUs) on board that work together, thus allowing the car to function fully, while also providing and improving the usability and comfort of the driver and passengers. In addition to control units, cars are composed of various sensors such as tire pressure sensors, tactile sensors that detect driver fatigue. Furthermore, modern cars tend to be increasingly connected to each other and to the automotive infrastructure.

For this reason, an increasing number of entities receive and transmit data across the ecosystem of connected vehicles. In particular, cars have different communication domains such as: vehicle-to-vehicle communication (V2V), Vehicle-to-Infrastructure communication (V2I), intravehicular communication (IV) and user-vehicle communication (U2V). Moreover, the car architectures are based on data and the latter are transmitted to the infrastructures via the communication domains.

Clearly, successful attacks in the automotive industry often combine social engineering practices with technical expertise. Hence, the manuscript focuses on the fundamental points of the automotive sector such as: cybersecurity, privacy and safety. In fact, the manuscript defines a security protocol called CINNAMON which aims to guarantee the confidentiality, authentication and integrity of intra-vehicular communications. Subsequently, the study concerning the privacy and trust of drivers is presented. A risk assessment exercise is then performed on eleven car brands. Finally, the automotive safety work carried out during the period abroad at Huawei is

presented regarding the development of an automatic safety analysis tool based on Papyrus models.

Acknowledgements

First of all, I thank my supervisor *Prof. Giampaolo Bella* for believing in me and for putting me in front of challenges that have been the result of an invaluable life experience. I would also like to give him all my esteem for giving me part of his critical sense. He has always pushed me towards excellence in a sincere and selfless way, giving me many opportunities to learn and grow professionally.

I would like to express my sincere gratitude to *Dr. Ilaria Matteucci* and *Dr. Gianpiero Costantino* for their availability and their advice in moments of indecision. Thanks again for teaching me not to give up and how to benefit from failed experiments.

I would also like to thank my two external reviewers, *Prof. Gianluca Dini* and *Prof. Bogdan Groza*, for helping me improve this thesis with their helpful feedback.

Thanks to *Ausilia*, who listens to me and supports me day after day. She has always pushed me towards self-realisation and I couldn't ask for better.

Thanks to my father *Agatino* and my sister *Federica*, they both believed in me and gave me a lot of loyalty and humility. Their support has been essential in achieving several milestones in my life.

My deepest recognition goes to my mother, *Adalgisa*, whose death helped me understand what really matters in life and what needs to be done to be happy. Thanks to her I understood that life being one and must always be lived with positivity and without regrets.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Modern cars offer highly developed technologies, such as infotainment systems and and e-call boxes, which are usually connected to the Internet. All this increases the possible attack surface, and there are a number of examples of remotely hijacked cars. Cars can also collect the personal data of drivers (or passengers), so privacy becomes a concern.

Intel estimates that a car can generate up to 4000 GB of data per day and has defined three groups of car data, depending on where the data comes from and how it is used [28]. The first group, which is called “inside-out”, includes data from vehicle sensors such as Light Detection and Ranging (LIDAR), motion sensors and video cameras. The second group, which is called “outside-in”, includes data from the vehicle’s external environment, e.g. traffic data, data from other vehicles and road infrastructure. It may be debatable whether such data qualifies as personal data of the driver, but the general position is that all data that can be associated, e.g. as intercepted by an attacker, with the driver as a natural person, qualifies as personal data.

For this reason, it is very important to understand and collect what types of (personal) data categories cars are collecting - and their manufacturers are processing - in particular if these include special categories according to Regulation (EU) 2016/679, known as GDPR [49]. Furthermore, it is important to understand whether and to what extent motorists are aware of what and how much data they disclose to car manufacturers and how this data is handled by them [19].

Despite some recent headlines about attacks on real cars [89], there is limited literature demonstrating how drivers feel about their privacy in their cars and what level of trust they place in e.g. the interconnected infotainment systems that are becoming increasingly common today.

1.2 Modern car architecture

Modern cars have many different types of electronic control units (ECUs) on board that work together, thus enabling the complete operation of the car, they also provide and improve the usability and comfort of drivers and passengers. The communication between the ECUs takes place via the CAN-Bus, which was developed by Bosch in 1983 and is standardised in ISO 11898-1:2015 [74] as a simple protocol based on two bus lines. Unfortunately, this protocol was designed without any security in mind. In fact, it turns out to be the main attack vector for most attacks on cars. The main access point to the CAN-Bus is the OBD-II port commonly used for vehicle diagnostics or to extract data regarding the car's location and movements. It seems clear that the car is actually composed of different types of hardware working together.

In addition to control units, cars are composed of different sensors such as tyre pressure sensors, touch sensors that detects driver fatigue through grip, pulse or temperature sensors in the passenger compartment. Other sensors are installed above the roof of the car, such as radar or exterior cameras that take the car into the world of autonomous driving. In fact, modern cars tend to be increasingly connected to each other and to automotive infrastructures. This connection can be made through the use of a SIM card, which provides a connectivity point for the transmission of information on board and in doing so allows the car to connect to the internet. Clearly, all this connectivity between the infrastructure and the car increases the risk of data breaches or attacks on cars.

In this respect, the automotive field can be divided into communication domains that process and manage different types of data as shown below [19].

1.2.1 Communication domains

Modern cars appear to be increasingly complex and connected systems. An increasing number of entities receive and transmit data through the connected vehicle ecosystem. In particular, cars have several communication domains such as:

- **Vehicle-to-Vehicle (V2V)** communication includes a wireless network where cars exchange messages with information about what they are doing. This data includes speed, position, direction of travel, braking and loss of stability. The aim of V2V communication is to prevent accidents by allowing passing vehicles to exchange position and speed data via an ad hoc network.

- **Vehicle-to-Infrastructure (V2I)** communication is a communication model that allows vehicles to share information with the components that support a country's highway system. Such components include cameras, traffic lights, lane markers. Therefore, sensors can capture infrastructure data and provide travellers with real-time alerts on issues such as road conditions, traffic congestion and parking availability.

- **Intra-Vehicle communication (IV)** is the communication model in which the ECUs and sensors communicate with each other and exchange messages about the car's status. All this allows a car to function properly.
- **User-to-Vehicle communication (U2V)** concerns the inclusion of the smartphone, its data and functionalities (e.g. to allow easy hands-free dialling of the address book) via Bluetooth, Wi-Fi or USB mirroring.
- **Car maker and third-party services** need to collect data. The transmission of data to these services takes place through the use of a SIM card inside the car that provides connectivity to the car. In this scenario, the car transmits data both to the car maker's servers and to third-party services (e.g. for software updates or with satellites for geolocation), which are often accepted during the acceptance of privacy policies.
- **Emergency services** is one of the on board services. Thanks to the car's built-in SIM, the car can call the emergency services in case of a rescue need, such as a brutal impact that caused the airbags to deploy.

The information about a modern car, the data it handles and transmits, and the components from which it is made can be summarised in the Figure 1.1.

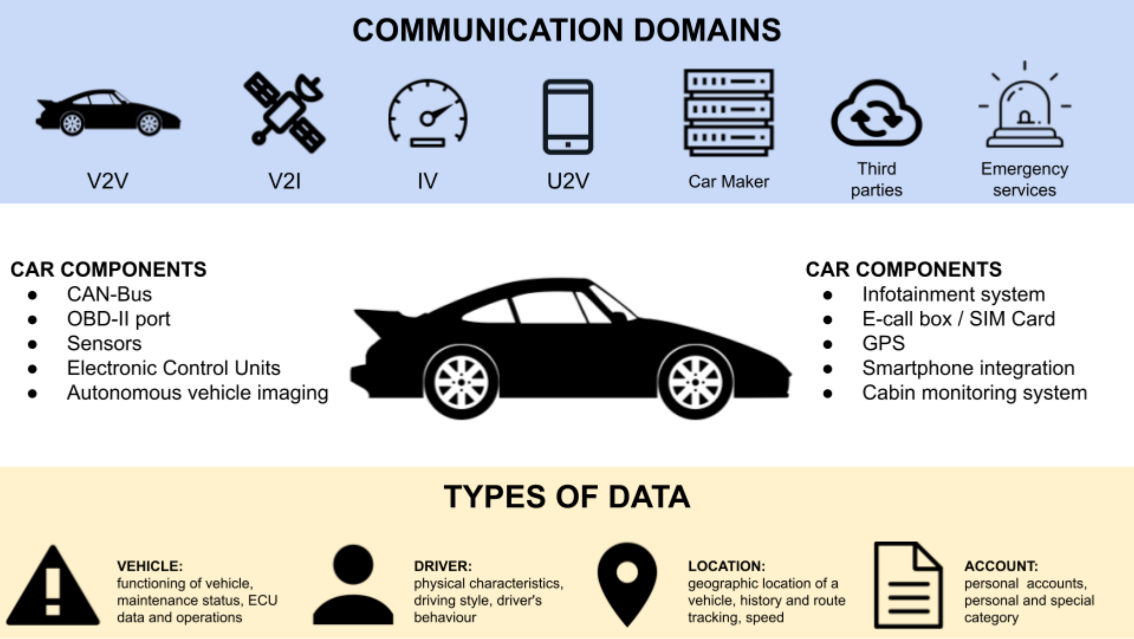

COMMUNICATION DOMAINS
V2V
V2I
IV
U2V
Car Maker
Third parties
Emergency services
CAR COMPONENTS
CAN-Bus
OBD-II port
Sensors
Electronic Control Units
Autonomous vehicle imaging
CAR COMPONENTS
Infotainment system
E-call box / SIM Card
GPS
Smartphone integration
Cabin monitoring system
TYPES OF DATA
VEHICLE: functioning of vehicle, maintenance status, ECU data and operations
DRIVER: physical characteristics, driving style, driver's behaviour
LOCATION: geographic location of a vehicle, history and route tracking, speed
ACCOUNT: personal accounts, personal and special category

Figure 1.1: Infographics about data receivers, data type and car components [19]

1.2.2 Data treatments

The car architectures are based on data and the latter are transmitted to the infrastructures via the communication domains. All these data can be classified into several types [19]:

- **Vehicle Data** are all data concerning vehicle operation, maintenance status, mileage, tyre consumption, data from sensors.
- **Driver Data** relates to driver profiling. This can be done by sampling the physical characteristics or habits of the driver, such as, driving style of the vehicle, seat belt use, braking habits.
- **Location Data** includes data, such as the geographical position of a vehicle, route history and tracking, speed, direction of travel.

- **Account Data** contains all data relating to the driver's personal accounts. An example would be when the driver connects his preferred music provider account with the car. This type of data in the GDPR is identified as a special category.

Furthermore, it can be noted that data from all the categories defined above, when intertwined with each other (by inference) can give rise to new information, allowing finer and more detailed profiling of the driver.

Processing data provides a number of benefits to drivers. Vehicle sensors and internal components can produce data for diagnostic purposes to check the health of the vehicle. Predicting and preventing component failures can effectively improve driver safety. In addition, processing sensor data allows technicians to conduct remote diagnostics and, by analysing historical data, repair costs can be lower. The collection of data can also be useful in emergency situations. In fact, all cars sold in Europe since April 2018 must implement an emergency-call system, which is called eCall, as a measure to reduce fatalities caused by road accidents [48]. The eCall system is an electronic device that automatically alerts the emergency services in the event of a car accident. This system makes rescue operations faster and more effective and helps to save lives. When a severe impact is detected via the motion sensors, using the GPS and the vehicle's built-in SIM card for communication, data such as the exact position of the crashed car, direction of travel and type of impact are sent to the emergency services. This communication can also be activated manually by the driver by pressing a button, thus indicating that he is alive and alert.

External service providers can use data to offer benefits to drivers, including economic benefits. These include usage-based car insurance, also known as pay-as-you-drive insurance. Insurers monitor the driving habits through a telematics device installed on the vehicle. By analysing data such as mileage, hard braking, rapid accelerations and cornering, the insurer is able to adapt the premium amount based on the driver's behaviour. The choice of a usage-based insurance over a traditional one has many potential benefits to the drivers. Thanks to the telematics device, drivers can monitor their driving habits and may get motivated to correct them because good drivers benefit from discounted premiums [97]. It is also easier to investigate in car accidents by having access to the events that took place before the accident occurred.

As previously announced, the infotainment system interacts directly with the driver and passengers by providing them with various functions, but these require data from users in order to function properly. By synchronising personal devices with the infotainment system, users are able to stream multimedia files, make calls using the car's speakers, read e-mails and send text messages. They can also surf the web using the system's browser. The infotainment system can provide directions and traffic information using the vehicle's real-time location. Compared to sensors, this is the main component that makes use of the personal data of its users.

Another type of data concerns the personal data of drivers and passengers, including music preferences, favourite places, contact list, text messages and call logs. In particular, it can be noted that this group may include "special categories of personal data", i.e. data on health, political and religious orientation, biometric data, ethnic origin, as defined by Article 9 of the GDPR. For example, the car manufacturer may infer certain health conditions of the driver by intersecting historical data from seat weight sensors with the chosen interior temperature, seat back adjustment, car position or heart rate via sensors on the car steering [2]. Remarkably, this data set can also include the driver's financial data to pay for fuel and parking [98].

Thus, through vehicle geolocation we may be able to obtain information that is repeated over time and thus derive a possible habit, hobby or routine of a driver. For example, we could notice that every Sunday at a certain time the car is near a church, so we could deduce that the driver goes to church every Sunday and through the information obtained in an open way from the network we are able to deduce also the information regarding the type of religion that the driver professes.

Finally, more and more experiments show that cars can also manage the driver's personal data, often transmitted in the infotainment network, such as driving style [91], location history [23] and also more general data such as cabin preferences, music preferences, and credit card details [50]. However, the literature shows how the infotainment system can provide an entry point for attackers. Some examples involve exploiting vulnerabilities in the infotainment of a General Motors car to steal data from the remote system [76]. Additionally, a few years ago, researchers discovered a number of vulnerabilities that, when combined, allowed them to remotely hack a Tesla Model S [37].

1.3 Objectives

The need to control modern cars from different points of view is therefore strongly justified. For this reason, useful objectives are defined below to allow the improvement of the automotive world.

- **Objective 1:** Understand all types of data and communication domains that affect the automotive world in order to understand how to reduce weak points;

- **Objective 2:** Define and implement software, in accordance with current guidelines, that provides the main security properties to automotive communication protocols;
- **Objective 3:** Understand driver privacy and trust issues;
- **Objective 4:** Understand the risk of the automotive world;
- **Objective 5:** Understand and analyse the automotive world also from a safety analysis perspective.

1.4 Contributions

The aim of this thesis is to study the different pivotal points of the automotive field. In particular, we focus on the point of view of cybersecurity in intra-vehicle communications, then on the privacy and trust of drivers, followed by a study of risk through the exercise of risk assessment and finally, on the point of view of safety in the automotive field. To achieve this, we assume a bottom-up approach to these fundamental points of the automotive field. Specifically, with regard to the cybersecurity of intra-vehicle communications, a module for secure communication between the ECUs of the car has been developed. With regard to driver privacy and trust, a crowdsourced study has been carried out to obtain drivers' perceptions and concerns. This was followed by a risk assessment exercise also carried out on several car brands. Finally, an automatic safety analysis tool based on Papyrus [54] models was developed.

Based on the previously defined objectives, the main contributions of this thesis are the following:

- **Contribution 1:** The definition of communication domains and data types and their intertwining and use in the automotive field.
- **Contribution 2:** Definition and implementation of CINNAMON, a basic software module based on AUTOSAR that aims at confidentiality, integrity and authentication, for the traffic exchanged on the bus protocols supported by AUTOSAR.
- **Contribution 3:** Statistics on car drivers' privacy concerns and trust perceptions obtained through crowdsourcing
- **Contribution 4:** The formulation and execution of the cybersecurity risk assessment per car brand.
- **Contribution 5:** The definition and development of an automatic safety analysis tool based on Papyrus models

The contributions of this thesis have been published and submitted to international journals and conferences:

Conference Giampaolo Bella, Pietro Biondi, Gianpiero Costantino, Ilaria Matteucci. "CINNAMON: A Module for AUTOSAR Secure Onboard Communication". In *16th European Dependable Computing Conference (EDCC 2020).* IEEE. 2020 [17].

Conference Giampaolo Bella, Pietro Biondi, Gianpiero Costantino, Ilaria Matteucci, Mirco Marchetti. "Towards the COSCA framework for "COnseptualing Secure CArs"". In *Open Identity Summit 2021.* Gesellschaft für Informatik. 2021 [20].

Conference Giampaolo Bella, Pietro Biondi, Marco De Vincenzi, Giuseppe Tudisco. “Privacy and modern cars through a dual lens”. In *IEEE European Symposium on Security and Privacy Workshops (EuroS&PW 2021).* IEEE. 2021 [19].

Conference Giampaolo Bella, Pietro Biondi, Giuseppe Tudisco. “Car Drivers’ Privacy Concerns and Trust Perceptions”. In *International Conference on Trust and Privacy in Digital Business (TRUSTBUS 2021).* Springer, Cham. 2021 [15].

Conference Giampaolo Bella, Pietro Biondi, Fabrizio Tronci. “Papyrus-based safety analysis automatization”. In *6th International Conference on System Reliability and Safety (ICSRS).* IEEE. 2022 [60].

Journal Giampaolo Bella, Pietro Biondi, Gianpiero Costantino, Ilaria Matteucci. “Designing and Implementing an AUTOSAR-based Basic Software Module for Enhanced Security”. *Computer Networks.* Elsevier. 2022 [18].

Journal Giampaolo Bella, Pietro Biondi, Giuseppe Tudisco. “A double assessment of privacy risks aboard top-selling cars”. *Automotive Innovation.* Springer. 2022 [14].

1.5 Thesis Outline

The thesis is divided into 6 chapters, plus 2 appendices. Each chapter deals with a specific aspect of the themes investigated.

The remainder of the paper is organised as follows. Chapter (§2) explains all the cybersecurity part applied to the automotive and describes CINNAMON: A Module for AUTOSAR Secure Onboard Communication. Chapter (§3) explains the work regarding the privacy and trust of drivers. Chapter (§4) deals with the study on risk assessment applied to various car brands. Chapter (§5) explains the work done on the functional safety applied to embedded systems. Finally, the document concludes with some broader evaluations of the results (§6).

Chapter 2

CINNAMON: A Module for AUTOSAR Secure Onboard Communication

An increasing number of Electronic Control Units (ECUs) communicate with each other to accomplish the functionalities of modern vehicles. ECUs form an in-vehicle network that is precisely regulated and must be adequately protected from malicious activity, which has had several outbreaks in recent years. Therefore, this chapter presents CINNAMON, an AUTOSAR-based Basic Software Module that aims at confidentiality, integrity and authentication, all at the same time, for the traffic exchanged over the bus protocols that AUTOSAR supports. CINNAMON in fact stands for Confidential, INtegral aNd Authentic onboard coMmunicatiON.

This Chapter introduces the requirements and specification of CINNAMON in a differential fashion with respect to the existing *Secure Onboard Communication* Basic Software Module, which does not include confidentiality. As a result, CINNAMON exceeds SecOC at least against information gathering attacks. The Chapter then defines three security profiles, regulating also the freshness attribute appropriately. Most importantly, CINNAMON is not a simple academic exercise because it is implemented in a laboratory environment on commercial ECUs, thus reaching the level of TRL 4, “Component and/or breadboard validation in laboratory environment”. The runtimes obtained on inexpensive devices are reassuring, paving the way for a possible large-scale application.

The Chapter is structured as follows: Section 2.1 outlines useful background notions. Section 2.2 introduces the assumed threat model. Section 2.3 presents the requirements of CINNAMON and Section 2.4 its specification and integration in AUTOSAR Classic Platform. Section 2.5 details the CINNAMON Security profiles. Section 2.6 discusses the prototype implementations. Section 2.7 describes the main parts of the code. Section 2.8 shows the runtimes obtained through our experiments. Section 2.9 treats the related work. Section 2.10 summarises and discusses all the contents of the chapter.

2.1 Background

This Section provides the essential background notions that will be useful in the sequel of the manuscript.

2.1.1 CAN vs Ethernet

Nowadays, cars are equipped with electronic components that communicate with each other through an internal network. Typically a modern car includes several dozen control units called Electronic Control Units (ECUs). These devices are connected to sensors and actuators and are used to carry out a large number of operations such as: seat heating, rearview and side mirror adjustment, engine temperature control, speed display on the digital speedometer The increase in the number of control units in cars implies higher cabling requirements, for this reason buses are used that are capable of connecting and controlling the various control units in the car efficiently, reliably and in real time. For many years the CAN bus (Controller Area Network) has been used for its cost efficiency. However, Ethernet looks like a possible future replacement for CAN due to its bandwidth efficiency, although it is more expensive and production costs are expected to decrease over the years. This section examines two automotive communication standards, CAN bus and Ethernet, in order to understand who is best suited to the automotive system and its use [109].

Controller Area Network

The Controller Area Network (CAN) is a serial standard of the multicast type [1], mandatory in vehicles since 2001. CAN-bus was introduced in the 1980s by Robert Bosch GmbH, to connect several ECUs together. Even today the CAN-bus is the protagonist in the automotive sector, since it was expressly designed to work without problems even in environments strongly disturbed by the presence of electromagnetic waves. In addition, CAN allows controllers, sensors and actuators to communicate with each other at a rate of up to 1Mbit/sec, while also offering: low design costs, ease of configuration and automatic detection of transmission errors. The CAN works on two wires: HIGH (CANH) and LOW (CANL) and uses the differential signal, this means that: when a signal arrives, the CAN increases the voltage on one line and reduces the voltage in the other line. an equal measure. The differential

[1] A receiving node can check the message content and filter only the packets it is interested in, ignoring the others.

signal is used in environments that must be fault tolerant, which is why it is used in the automotive field. The reduction of electromagnetic disturbances can be further increased by using twisted pair cables.

The CAN-bus uses Carrier Sense Multiple Access with Bitwise Arbitration (CSMA/BA) [131] which is a multiple access protocol to communication channels. CSMA/BA is the evolution of the CSMA/CD data link layer protocol, created to ensure that transmission conflicts, i.e. collisions, present both in pure CSMA and in CSMA/CD are no longer destructive for messages . If two or more devices start transmitting at the same time, a priority-based arbitration mechanism is applied to decide which device to allow to continue transmitting. This priority is assigned to each node with a sequence of bits to be sent at the beginning of the communication. During the arbitration phase, each transmitting node checks the bus status and compares the bit received with the bit transmitted, through the bitwise AND operation. If a dominant bit (D bit - logical 0) is received while a recessive bit (R bit - logical 1) is being transmitted, the node that sent the recessive bit stops the transmission, i.e. loses the arbitration. If, on the other hand, the node does not notice variations between the bits received and those transmitted, it will win the "challenge" and even if a collision has occurred, this will not destroy the winner's message. At the end of the sending of the priority bits, all the nodes that have lost the arbitration suspend the transmission and retry later, in this way the message of the node with the highest priority can pass freely in the communication channel. This mechanism guarantees that at the end of the contention phase only one station remains active and in so doing it will complete the transmission by sending the message.

Since the CAN can be shared by many ECUs, it is generally not very suitable for sending messages that may require updates more than 100 times per second. Ideally it is suitable for transmitting much slower status updates. This includes applications such as communication between mechanical systems (transmission, braking, cruise control, windows, locks) where the amount of data is limited and the bandwidths involved tend to be relatively low.

Messages are commonly named *CAN frames* and each frame contains various fields. These include an Arbitration field carrying the frame ID, also used for arbitration, a Control field for control signals and a Data field for the payload. The latest version is CAN2.0, dating back to 1991. The 2.0A version carries an 11 bit frame identifier, as pictured in Figure 2.1. The full CAN frame format is described below.

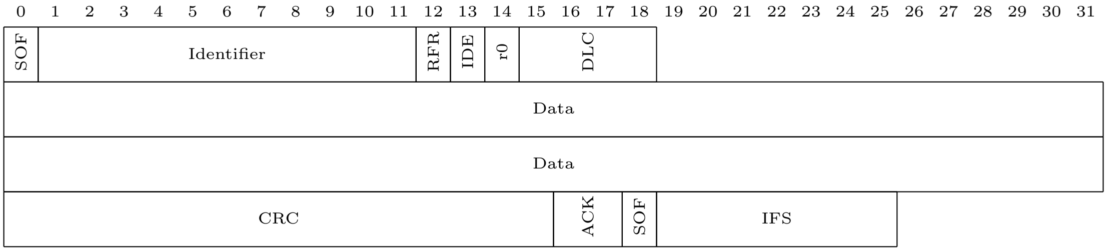


Figure 2.1: Standard CAN 2.0A frame format

- **Start Of Frame (SOF)** is a dominant bit indicating the beginning of a frame.
- **Arbitration field** consists in:
  - *Identifier*, 11 bits (CAN 2.0A) or 29 bits (CAN 2.0B), to signify the priority of the frame, with a lower value indicating higher priority.
  - *Remote Transmission Request* (RTR), 1 bit, which is low (dominant) for a Data Frame and high (recessive) for a Remote Frame (one whose Data Field is empty).

- **Control field**, includes
  - *the IDE field*, 1 bit, to identify whether the payload is of standard length;
  - *r0*, 1 bit, reserved for later use;
  - the *Data Length Code (DLC)* field, 4 bits, indicating the length of the Data Field.
- **Data** spans over up to 64 bits of data, and carries the payload of the frame.
- **CRC**, 15 bits, is for a cyclic redundancy check code and a recessive bit as a delimiter.
- **EOF**, 7 bits, all recessive, indicates the end-of-frame.
- **IFS**, 7 bits, indicates the time for the controller to move a correct frame into the buffer.

Ethernet in automotive

Ethernet is one of the most common high-speed interfaces found in homes and offices, and there are some cars where Ethernet is used to carry a variety of high-speed data. Like CAN, Ethernet is a packaged system, where information is transferred in packets between nodes on various parts of the network. Also, like CAN, Ethernet is bidirectional and the possible speed on each individual link decreases as the number of nodes on the system increases. However, Ethernet can carry data over a link 100 times faster than CAN. Ethernet is useful for medium bandwidth communications in applications such as navigation and control systems. It can be used in much the same way as CAN while providing much more bandwidth, the difference being that

Ethernet requires a star topology and a gateway. Ethernet would be an ideal choice to replace CAN, but because the cost per node of Ethernet is higher, it probably won't replace, but rather can lead to increased CAN utilisation.

For example, to transport video from a parking camera over an Ethernet network, the video must be compressed at the source, then decompressed at the destination to avoid exceeding Ethernet bandwidth limits. For this type of application it is necessary that the camera must have a relatively high-powered processor to compress the image sufficiently to insert it into the Ethernet network. A disadvantage of this solution is that video compression and decompression add latency to the link.

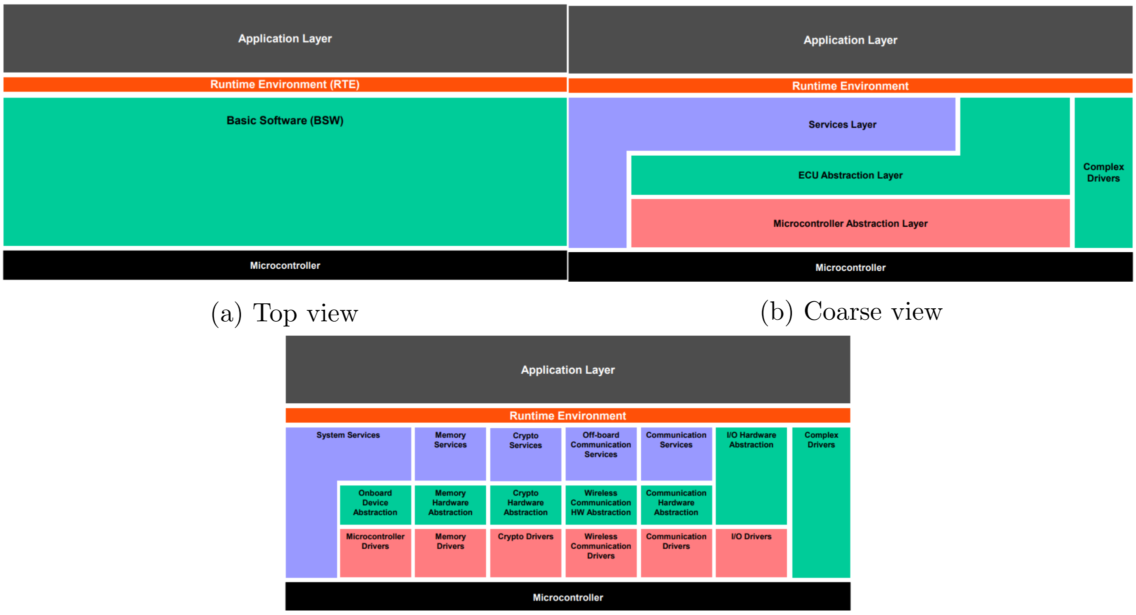
2.1.2 AUTOSAR

AUTomotive Open System ARchitecture (AUTOSAR) is a worldwide development partnership of vehicle manufacturers, service providers and companies in the automotive electronics, semiconductor and software industries founded in 2003. The AUTOSAR partnership is founded by nine companies with the aim of consolidating company skills and defining an open automotive system architecture standard to support the needs of future car applications. AUTOSAR core partners are: BMW Group, Bosch, Continental, Daimler, Ford, General Motors, PSA Group, Toyota Volkswagen Group. More specifically, some of AUTOSAR's objectives are: to create and establish an open and standardised software architecture for automotive electronic control units (ECU). Goals include scalability to different vehicle and platform variants, software transferability, consideration of availability and security requirements.

The AUTOSAR architecture is divided into three software levels which run on a microcontroller [7]. The software levels are called *Application, Runtime Environment* and *Basic Software* as shown in Figure 2.2a. The Basic Software is divided into levels: *Services, ECU Abstraction, Microcontroller Abstraction* and *Complex Drivers* as shown in Figure 2.2b. The *Microcontroller Abstraction Layer* is the lowest software layer and contains drivers that have direct access to the microcontroller peripherals. The *ECU Abstraction Layer* interfaces the *Microcontroller Abstraction Layer* drivers. It also contains drivers for external devices and offers APIs for accessing peripherals and devices. The *Services Layer* is the highest level of the *Basic Software* and offers: operating system functions, communication services and vehicle network management, memory services. Finally, the Basic Software is further subdivided into functional groups as shown in Figure 2.2c. Thus, the Basic Software can be divided into the following types of services:

- Input/Output: Access to sensors, actuators and on-board peripherals.
- Memory: Access to memory.
- Crypto: Access to cryptographic primitives.
- Communication: Access to vehicle network systems, ECU communication systems.
- Off-board Communication: Access to *Vehicle-to-X* communication, vehicle wireless network systems, external communication systems with ECU.
- System: It provides services to the system such as: timer, error memory, ECU status management and library functions.

For each component of the architecture, AUTOSAR defines guidelines through requirements and specifications. Later we will see an application example.



(a) Top view

(b) Coarse view

(c) Detailed view

Figure 2.2: AUTOSAR architecture [7]

2.1.3 Key Management

As in any computer system, the management of the keys used for cryptographic operations must also be taken into account in the automotive field.

In the field of intra-vehicular communication, the aim is to use symmetric cryptography. AUTOSAR defines guidelines regarding Key Management and does so through the *Crypto Service Manager* (CSM) [8]. CSM is a service that provides cryptographic functionality, based on a cryptographic driver based on a software library or hardware module, as shown in Figure 2.3. In addition, mixed configurations with multiple cryptographic drivers are possible. Also from the figure, we can see that at the bottom of the stack concerning the CSM we find the Security Hardware Extension (SHE) and Hardware Security Module (HSM). Both are specialised hardware capable of calculating the appropriate algorithms, in parallel with the main processor. The CSM and associated cryptographic libraries then pass requests to these hardware and check cyclically if a result is available. The HSM normally has its own protected RAM, an exclusive flash area for code, and its own peripherals such as timers, hardware accelerators for some cryptographic algorithms or generators for random numbers. This allows for implementation of secure, authenticated system boot or host monitoring at runtime. The unique data flash memory can be used to store secrets in such a way that they are not accessible from the host system. This means that the host can request a cryptographic operation performed by the HSM without the key leaving the HMS. For this reason, a common solution in the automotive field is to keep a *pre installed key* that is a key inserted inside the HSM at the time of construction in the factory. There are several examples of HSM, one of the most famous is ARM TrustZone [6]. Using TrustZone, the processor can execute instructions in one of two possible security modes, called “normal”, in which untrusted code is executed, and “secure”, in which protected services are executed. These processor modes have independent memory address spaces and different privileges.

2.1.4 SPECK64/128 and CHASKEY MAC

SPECK64/128 is a lightweight block cipher publicly released by the NSA in 2013 [10]. It is an Addition-Rotation-XOR (ARX) cipher and works with a 128 bits key on blocks of 64 bits. It has been designed to have the full security possible for each block and key size against standard chosen-plaintext (CPA) and chosen-ciphertext (CCA) attacks. According to the literature, no successful attack on full-round SPECK64/128 (27-rounds) is known. By contrast, reducing the number of rounds invites *differential cryptanalysis attacks* in the standard attack model (CPA/CCA

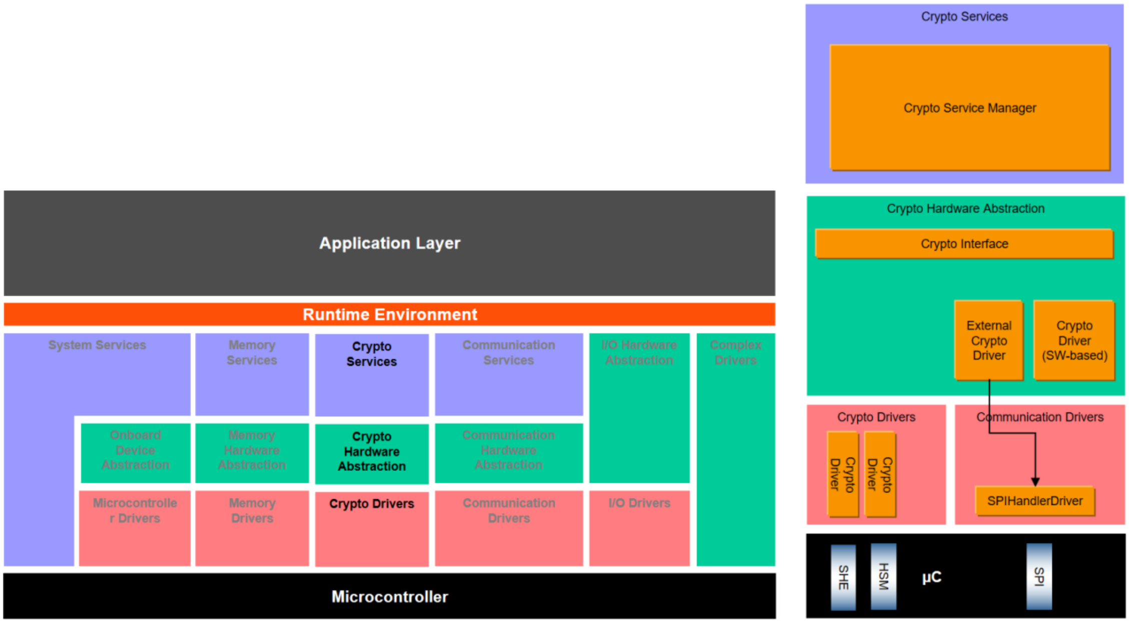


Figure 2.3: AUTOSAR architecture with CSM

with unknown key), which succeed when rounds are reduced to approximately 70-75%; still, such attacks are only marginally faster than brute-force [45].

Chaskey MAC Chaskey is a Message Authentication Code (MAC) algorithm whose design is thought for 32-bit micro-controller architectures [95]. The addition and XOR operations are performed on 32-bit words, and each of these operations requires only one instruction on the target architectures. We are aware of the inherent risks of tag guessing (finding counter-images) or tag collision (birthday attacks) for every MAC [46, 130]. However, Chaskey will be used below to produce a tag size of 128 bits, a choice that largely satisfies the official recommendation of exceeding 64 bits. Also, because “Chaskey is robust under tag truncation” [95], it seems safe to truncate each calculated tag as prescribed by SecOC [9], for example to 24 bits, thus complying with AUTOSAR.

2.2 Threat Model

This thesis assumes a threat model with an active attacker who may exploit some vulnerabilities of in-vehicle network to gain some digital access to the car, either locally or remotely. More precisely, our attacker attempts to carry out two sets of malicious activities:

- **Malicious activity set 1**, the injection of frames she altered or built from scratch using known data, thus manipulating the data processed by the target ECUs to trigger specific functionalities. This activity set includes:
  - *tampering*, the manipulation of frames with the aim of invalidating their contents so that receiving ECUs cannot perform execute the operations that were originally meant;
  - *fuzzing*, the manipulation of frames with the aim of studying the behaviour of target ECUs;
  - *forging*, the generation of a valid frame with the aim of generating a valid signal and activating a specific ECU functionality;
  - *replaying*, the reuse of valid frames with the aim of repeating the generation of a valid signal and reactivating a specific ECU functionality.
  - *masquerading*, the generation of a valid frame with the aim of abusing the identifier of another, genuine ECU.
- **Malicious activity set 2**, the collection of information about the running protocols and other mechanisms in place in the network she observes, in particular acquire exchanged data. This activity set includes:

  - *sniffing*, the capture of frames in transit with the aim of learning the data they carry;
  - *information gathering*, the capture of frames in transit with the aim of identifying and interpreting the full set of data they carry, such as payloads and their associated ECU functionalities.

The attacker's malicious activities are clearly related to the security properties and attributes that we would like to establish, in the sense that each set of malicious activities attempts to undermine a set of security properties and, in turn, that set of security properties attempts to thwart the given set of malicious activities. In particular:

- Malicious activity set 1 relates to security property and attributes:
  - *authentication*, the identity of the sender of a given frame can be verified;
  - *integrity*, the content of a given frame is not altered during transmission;
  - *freshness*, it can be verified whether a given frame was already received.
- Malicious activity set 2 relates to security property set:
  - *confidentiality*, the content of a given frame is not disclosed to unauthorised entities.

Our attacker is limited in the sense that she only has partial control of ECUs, hence she cannot:

- obtain privileged access to any ECUs and, in particular,

- access the keys to be used for MACs and cryptographic operations, which are assumed to be deployed on each ECU and protected by a Crypto Service Manager (CSM) or similar solutions.

2.3 CINNAMON Requirements

This section introduces the requirements of CINNAMON and positions the new module within the AUTOSAR Classic Platform. CINNAMON inherits most of its requirements from AUTOSAR Secure On-Board Communication module (SecOC) without modifications, hence those are not discussed here. By contrast, CINNAMON inherits and modifies some requirements from SecOC in order to account for encryption. Such new requirements are presented below with the caveat that the modifications determined by the new module are highlighted in boldface; also, the new requirements coherently preserve the numbering of the old ones.

2.3.1 Functional Requirements

Requirement CINNAMON_00003 (Table 2.1) configures different security properties, with the noteworthy inclusion of confidentiality. In, essence, it prescribes that security experts be able to define the level of protection of onboard communication messages and the parameters needed to configure the functionalities of the module, including encryption and decryption.

2.3.2 Initialisation Requirements

Requirement CINNAMON_00005 (Table 2.2) initialises the security information and also covers encryption. In fact, the requirements explicitly refers to encryption

Table 2.1: CINNAMON_00003 requirement

Functional Requirement	*Configuration of different security properties/requirements*
Type:	Valid
Description:	Different security properties**, notably including confidentiality,** shall be configurable.
Rationale:	The assessment may vary in several parameters and its security needs. Thus the level of protection shall be configurable to adapt to these needs by means of a set of adequate parameters.
Use Case:	Security experts define the different security properties. For every message with security protection needs, the appropriate properties may be selected.
Corresponding SecOC Requirement	SRS_SecOC_00003

initialisation parameters and keys needed for the encryption phase. Such phase aims at including confidentiality among the set of CINNAMON security properties.

Requirement CINNAMON_00030 (Table 2.3) inherits from SecOC that it shall be possible to extract the payload from secured messages without authentication and insists on this to be possible even if the payload is encrypted.

2.3.3 Non-Functional Requirements

Requirement CINNAMON_00025 (Table 2.4) refers to performance, thus it regulates the computation time to perform security operations, in particular for dealing with cipher-texts.

Table 2.2: CINNAMON_00005 requirement

Initialisation	*Initialisation of security information*
Type:	Valid
Description:	The CINNAMON module's security configuration shall get initialised at module start-up.
Rationale:	The CINNAMON module needs security configuration information (Key-IDs **for calculating MACs**, Freshness Values, **encryption initialisation paramaters and Key-IDs for encryption**) to perform its operations. Therefore, this information shall get recovered and configured before it starts its processing operation.
Use Case:	CINNAMON loads the ID of the messages, the authorised authentication retry counter and the properties, **including confidentiality**, that are used for the processing of its incoming communications from upper and lower layers.
Corresponding SecOC Requirement	SRS_SecOC_00005

2.4 CINNAMON Specification

The CINNAMON module is a Software Base (BSW) module capable of protecting on-board CAN Bus communications and is based on AUTOSAR in the sense that it extends AUTOSAR in terms of possible functionalities provided and can be integrated into the current AUTOSAR architecture. CINNAMON is, as SecOC, part of the Communication Services of the AUTOSAR Classic Platform, as depicted in Figure 2.4. It encapsulates the SecOC module and inherits its API to interact with the Protocol Data Units (PDU) Router component and with the cryptographic services provided by the *Crypto Service Manager*. Also, our module interacts with the Run-Time Environment to manage counters and keys.

CINNAMON acts as a middle-layer between the low-layer communication module, i.e., TP, and the upper layer software module, i.e., AUTOSAR COM. In addition, our module internally manages the communication with the lower level to

Table 2.3: CINNAMON_00030 requirement

Normal Operations	*Support of capability to extract Authentic PDU without Authentication*
Type:	Valid
Description:	CINNAMON shall be capable to extract the payload from Secured frames, without Authentication **, even if the payload is encrypted**.
Rationale:	CINNAMON can be used as an extractor of payload from Secured frames, to enable low latency GW behaviour when a part of downstream communication clusters does not require authentication of frames.
Use Case:	Gateway.
Corresponding SecOC Requirement	SRS_SecOC_00030

build and send the secured data using a single PDU. Differently, the last version of SecOC specification [9] suggests to use two PDU, one dedicated to store information used to authenticate the sender of the frame, and another one containing the secured frame.

2.4.1 Authentication, Integrity and Confidentiality

CINNAMON inherits SecOC authentication and integrity mechanisms, reviewed in Figure 2.5.

AUTOSAR assumes that all ECUs have the cryptographic keys to handle Message Authentication Codes (MACs) (see [8]). Moreover, an external Freshness Value Manager provides counters to both sender and receiver to support the freshness of exchanged frames.

CINNAMON inherits the same prerequisites, briefly recalled here. Let us consider a sender ECU and a receiver ECU. Before sending a payload, the sender

Table 2.4: CINNAMON_00025 requirement

Non-Functional Requirements (Timing)	*Authentication and verification processing time*
Type:	Valid
Description:	Authentication, verification, **encryption and decryption** processing shall be performed in a timely fashion so that the real time critical signals do not get affected.
Rationale:	Transmission and reception of the time critical message between the running applications of two or more peers shall not get penalised by the additional processing of their underlying communication software layers such that the signals are finally rejected. It is necessary that when time critical messages are transmitted and received through secured messages, the additional processing required by CINNAMON remains under a value that is predictable and compatible with the time constraints of the concerned signals.
Use Case:	A legitimate **encrypted and** authenticated message is **decrypted,** verified and passed to the receiving ECU within the expected time-frame without experiencing signal monitoring errors.
Corresponding SecOC Requirement	SRS_SecOC_00025

generates the MAC starting from the payload and possibly the *Freshness Value* calculated according to the Monotonic Counter (Figure 2.5) provided by the Freshness Value Manager (an ECU may decide to ignore the Freshness Value). So, the secured frame is composed by the payload, the truncated MAC (MACT in Figure 2.5) and, optionally, the truncated freshness value (FVT).

The receiver has to validate the frame before accepting it and does this by verifying the MAC. In fact, the receiver generates a freshness value for verification (FVV) starting from the Monotonic Counter (Figure 2.5) received by the Freshness Value

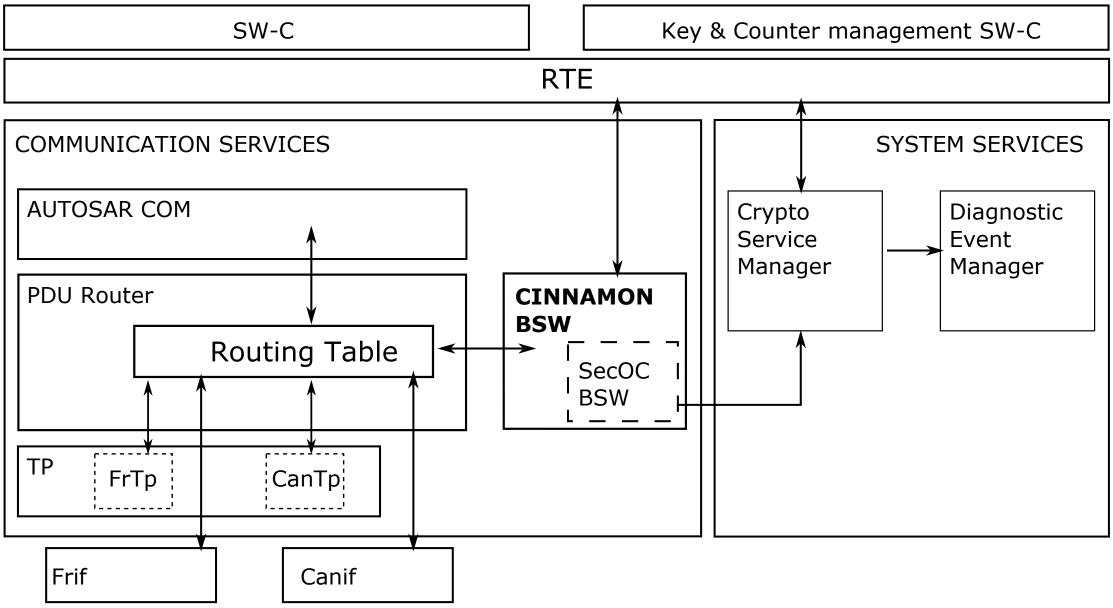


Figure 2.4: Integrating the CINNAMON BSW module within AUTOSAR Classic Platform

Manager and the previously received freshness value (the latest received counter in Figure 2.5). Then, it calculates the MAC by using the received payload and the FVV. If the outcome equals the received MACT, then the payload is accepted, otherwise it is discarded.

While MAC operations are normally fast, hence not problematic on inexpensive ECUs, it can be anticipated that the implementation of encryption and decryption primitives may cause a computational bottleneck. In consequence, the choice of the cryptographic scheme will have to be made with care.

Also, this specification purposely is not prescriptive with respect to the choice between the encrypt-then-MAC or MAC-then-encrypt approaches. This will be made at implementation level, depending on the adopted cryptographic scheme, its features and possible vulnerabilities. Another factor for this choice will be the specific transport protocol of the application scenario.

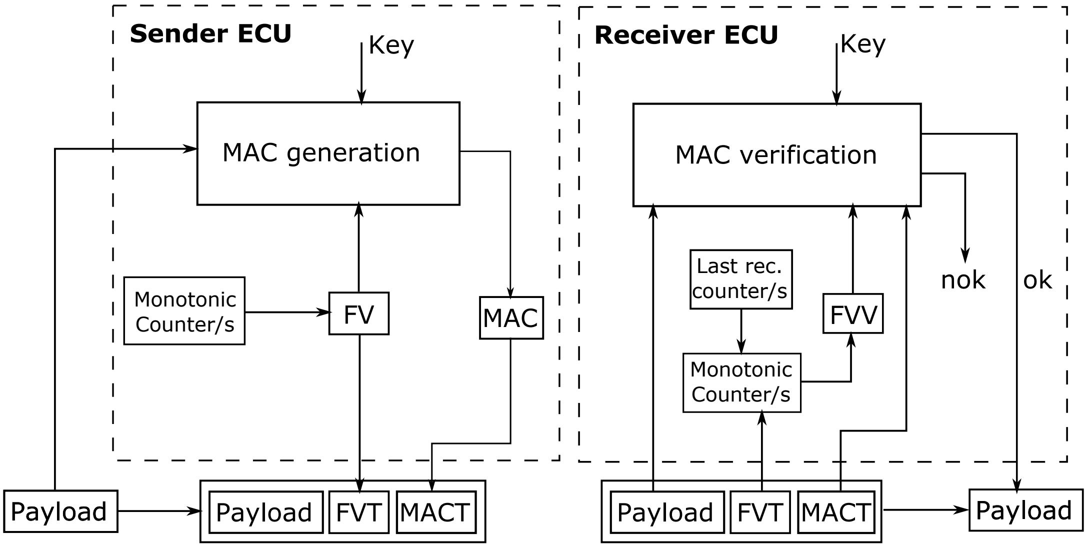


Figure 2.5: SecOC MAC Generation and Verification [9]

2.4.2 Freshness

The freshness value refers to a Monotonic Counter (MC), termed Freshness Counter (FC), which is used to guarantee the freshness of communication and must be provided by a Freshness Value Manager (FVM) unit which regularly distributes the freshness values on the network. In addition, the FVM takes care of synchronising and updating the freshness values in the ECUs appropriately. The FVM also allows the FC to be refreshed in a way that mitigates possible attacks such as replay and MiTM.

CINNAMON implements the freshness mechanism using two different ways that follows the AUTOSAR specifications. Note that, in total AUTOSAR describes three different approaches to building the freshness counter, two based on counters (single or multiple) and one on timestamps[2].

In the approach based on *Single Freshness Counter*, the FVM supplies the FV

[2] In this thesis, we do not consider the FV based on timestamp due to hardware constrains.

to nodes connected to the network and increases the counter each time a frame is sent in the communication channel. To guarantee freshness, the FCs of the sender and receiver should be increased synchronously. Thus, the FC must be incremented for each outgoing frame that has been validated on the receiving side.

The approach based on *Multiple Freshness Counters* uses four counters: *Trip Counter*, *Reset Counter*, *Message Counter* and *Reset Flag.* The FVM manages two of them, the Trip and Reset Counter. These counters are incremented according to specified criteria [9]. For example, the Reset Counter is incremented by 1 at regular time intervals. The multiple freshness counter approach requires the use of a `ClearAcceptanceWindow` parameter that represents an admissible freshness interval to counter potential lack of synchronisation.

Freshness Value Based on Multiple Counter

The generation and validation algorithms of FV based on Multiple Counter take into account four counters: *Trip Counter*, *Reset Counter*, *Message Counter* and *Reset Flag*. These counters are increased according to specific conditions provided by AUTOSAR [9]. In particular, the Trip Counter is increased by one when the FVM starts, resets and when the power status changes. The Reset Counter is increased by one at regular time intervals. The Message Counter is increased by one value for each message transmission. Instead, the Reset Flag is represented by the two least significant bits of the Reset Counter.

It is important to note that the all counters can assume the states of *"Latest"* and *"Previous"*, more specifically:

- Latest Trip Counter or Reset Counter refer to the values received from the FVM.

- Previous Trip Counter, Reset Counter, Message Counter and Reset Flag refer to the individual freshness values used for previous authentication generation or verification.

The above counters are stored in volatile memory with the exception of the Trip Counter which is stored in non-volatile memory to reduce data loss in the event of a sudden stop of the ECU. The FVM maintains only the Trip Counter and the Reset Counter that are initialised to 1. Instead, the ECU slaves use all four counters which are initialised to 0.

Generation phase The FV generation algorithm performs a comparison between the latest and previously sent value of both Trip Counter and Reset Counter. In particular, it checks whether the latest value is equal to the previously sent value of the two counters. Based on this comparison the FV is built. For simplicity, Table 2.5 shows the construction of the freshness value for the transmission. For example, if the comparison returns a positive result, then the FV is composed of the previously sent value as regards the Trip Counter and the Reset Counter, while the Message Counter will be equal to previously sent value plus one and the Reset Flag will be equal to the value from the lower end of the reset counter of the previously sent value as shown in Table 2.5.

Verification phase At verification, the FVV is built and used to validate the frame. The FVV is generated by performing three comparisons: Reset Flag, Trip Counter and Reset Counter and finally on the Message Counter. Based on the result of these comparisons, the value of the receiver's FV is constructed. For clarity, we report Table 2.6 (see [9]), which describes the algorithm on the possible scenarios. For example, if we consider the "Format 1" (first row of Table 2.6), we note that

Table 2.5: Construction of Freshness Value for Transmission [9]

Trip Counter and Reset Counter comparison	Construction of Freshness Value for Transmission			
	Trip Counter	Reset Counter	Message Counter	Reset Flag
Latest value = Previously sent value	Previously sent value	Previously sent value	Previously sent value +1	The value from the lower end of the reset counter (previously sent value)
Latest value ≠ Previously sent value	Latest value	Latest value	Initial Value +1	The value from the lower end of the reset counter (latest value)

the algorithm checks whether the latest value of the Reset Flag is equal to the received value. Then, it checks whether the latest value of the Trip Counter and Reset Counter is equal to the previously received value. Finally, the previously received value of Message Counter is checked to be less that the received value. If all the checks explained above are successful, then the FVV will be composed of the previously received value as regards the Trip Counter, the Reset Counter and the Message Counter (Upper), while the Message Counter (Lower) will be equal to the received value Message Counter Upper refers to the range that is not included in the truncated freshness value for Message Counter transmission, Message Counter Lower refers to the range that is included in the truncated freshness value for Message Counter transmission.

2.5 CINNAMON Security Profiles

As in the Secure On Board Communication module, also in the CINNAMON module it is possible to define and manage various security profiles. *Security Profiles* provide a consistent set of values for a subset of configuration parameters that are relevant for the configuration of CINNAMON. This is in line with the Secure Onboard Communication module [9]. A CINNAMON Security Profile is defined as the configuration of the following mandatory parameters.

- `algorithmFamily:String [0..1]` is the first parameter that characterises the used authentication algorithm. This parameter identifies the family of authentication algorithms.

- `algorithmMode:String [0..1]` is the second parameter that characterises the used authentication algorithm. This parameter identifies which MAC algorithm of the family is used.

- `algorithmSecondaryFamily:String [0..1]` is the third parameter that characterises the used authentication algorithm. This parameter identifies a secondary family of the chosen algorithm, if any.

- `authInfoTxLength:PositiveInteger` denotes the length of the truncated MAC.

- `freshnessValueLength:PositiveInteger` denotes the length of generated freshness value.
- `freshnessValueTruncLength:PositiveInteger` denotes the length of the truncated freshness value.
- `algorithmFreshnessValue:String [0..1]` denotes the algorithm used to generate the freshness value.
- `algorithmEncryption:String [0..1]` denotes the encryption algorithm.

Note that the first six parameters are inherited from SecOC, while the last two are typical of CINNAMON. This paper defines three example security profiles.

Security Profile 1 The CINNAMON Security Profile 1 is in Table 2.7. It does not set any parameters related to the FV and only uses the MAC and encryption algorithm parameters. These are set as Chaskey (§2.4.1) with `freshnessValueTruncLength` of 24 bits and SPECK64/128 (§2.4.1).

Table 2.7: CINNAMON Security Profile 1

Parameter	Configuration Value
`algorithmFamily`	Chaskey
`algorithmMode`	Chaskey_MAC
`algorithmSecondaryFamily`	not set
`SecOCFreshnessValueLength`	not set
`SecOCFreshnessValueTruncLength`	not set
`SecOCAuthInfoTruncLength`	24 bit
`algorithmFreshnessValue`	not set
`algorithmEncryption`	SPECK64/128

Security Profile 2 Security Profile 2 introduces a freshness value, so it is designed to avoid replay attacks on the communication channel, as shown in Table 2.8. As before, we define the parameters used by this profile with its respective values. It can be seen that FV is based on Single Counter, that `freshnessValueLength` is of 64 bits and `freshnessValueTruncLength:PositiveInteger` of 8 bits, so FVT is 8 bit long. Finally, the choices of MAC function and the cryptographic scheme remain the same as in Security Profile 1.

Table 2.8: CINNAMON Security Profile 2

Parameter	Configuration Value
`algorithmFamily`	Chaskey
`algorithmMode`	Chaskey_MAC
`algorithmSecondaryFamily`	not set
`SecOCFreshnessValueLength`	64 bit
`SecOCFreshnessValueTruncLength`	8 bit
`SecOCAuthInfoTruncLength`	24 bit
`algorithmFreshnessValue`	Single Counter
`algorithmEncryption`	SPECK64/128

Security Profile 3 Security Profile 3 is a modification of the previous one by means of a different method to generate the FV, as shown in Table 2.9. The method relies on *Multiple Counters* (see Section 2.4.2).

Table 2.9: CINNAMON Security Profile 3

Parameter	Configuration Value
`algorithmFamily`	Chaskey
`algorithmMode`	Chaskey_MAC
`algorithmSecondaryFamily`	not set
`SecOCFreshnessValueLength`	64 bit
`SecOCFreshnessValueTruncLength`	8 bit
`SecOCAuthInfoTruncLength`	24 bit
`algorithmFreshnessValue`	Multiple Counter
`algorithmEncryption`	SPECK64/128

Once the FV is generated, the sending ECU calculate the MAC of the payload and truncate the output by taking only 3 bytes. Also, the FV is truncated to 8 bits and assigned to the FVT variable, which is to be inserted in the payload.

On the other side, the receiver must perform the reverse operations. So, it first decrypts the received message, then executes the freshness value verification algorithm (see Section 2.4.2) to obtain a FV of 32 bits starting from the 8 bits of FVT. This operation is needed to check that the counters are synchronised. Once the FV is obtained, the receiver builds its temporary message to calculate the MAC and carry out the verification.

2.6 Implementations of CINNAMON

This section describes the prototype implementations of CINNAMON with its three security profiles on CAN bus.

At the implementation level, CINNAMON can be defined within a stack as shown in Figure 2.6. More in detail, starting from the bottom up, the stack is composed of:

- *Microcontroller* manages all the hardware part inside the car.
- *Driver* interface between the microcontroller and the Gateway ECU.
- *Gateway ECU* takes care of dividing the traffic into the subnets of the car.
- *Sub Networks* divided into:
  - **Infotainment:** whose components are Telephone, Navigation, Radio.
  - **Chassis:** whose components are Steering Control, Air bag Control, Breaking system.
  - **Power Train:** whose components are Power train sensors, Engine control, Transmission control.
  - **Body Control:** whose components are Instrument cluster, Climate control, Door locking.
- *Application*: application level where there are applications for the user.

Figure 2.7 shows an example of the stack application through a sequence diagram in which the layers perform operations in order to reach the properties provided by CINNAMON.

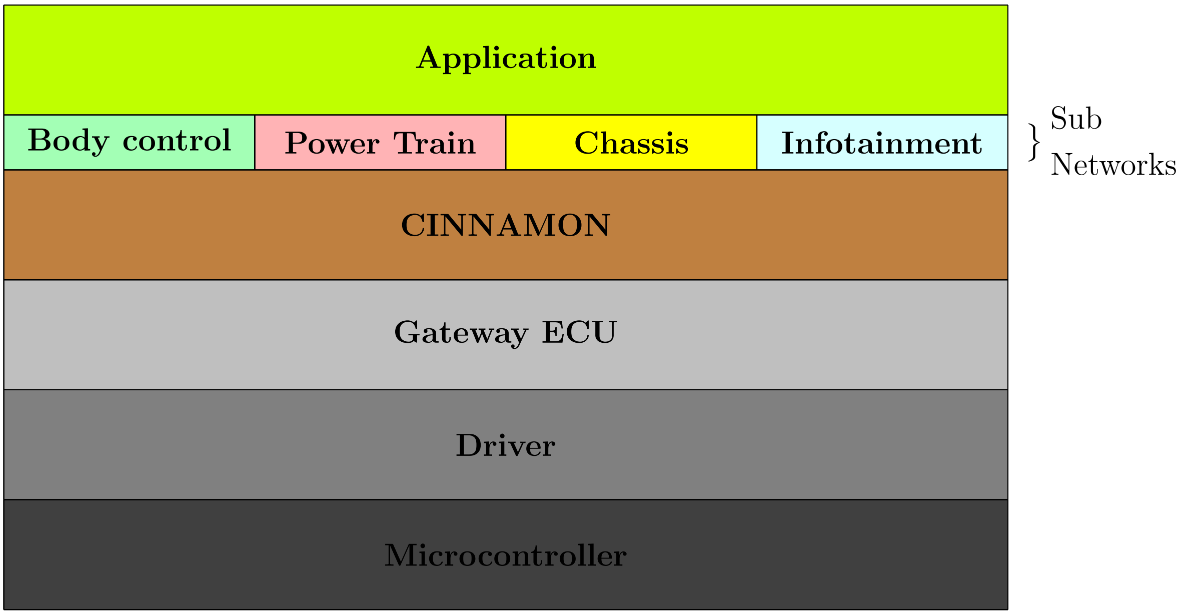


Figure 2.6: Stack design

2.6.1 Testbed

The testbed consists of two ECUs and a laptop interconnected via CAN bus. The laptop can send and receive frames in the channel, the other two are on STM32F407 Discovery boards connected to the laptop through a USB-to-CAN device. The boards come with an ARM Cortex M4 processor each, physical input buttons and light emitting diodes (LED) for visual outputs. Additionally, the boards are equipped with an additional STM32F4 DISCOVERY COMM shield to provide CAN bus connectivity.

The laptop acts as FVM in the experiments that are explained below, and it is assumed that all participating ECUs are provided with the necessary cryptographic keys. The laptop runs software *IXXAT canAnalyser 3* [75] to manage the USB-to-CAN device, while the two boards run our code discussed in the sequel of this manuscript. Moreover, we can input to canAnalyser 3 specific frames that we want the laptop to send, and this is useful to test broadcast reception, namely by both

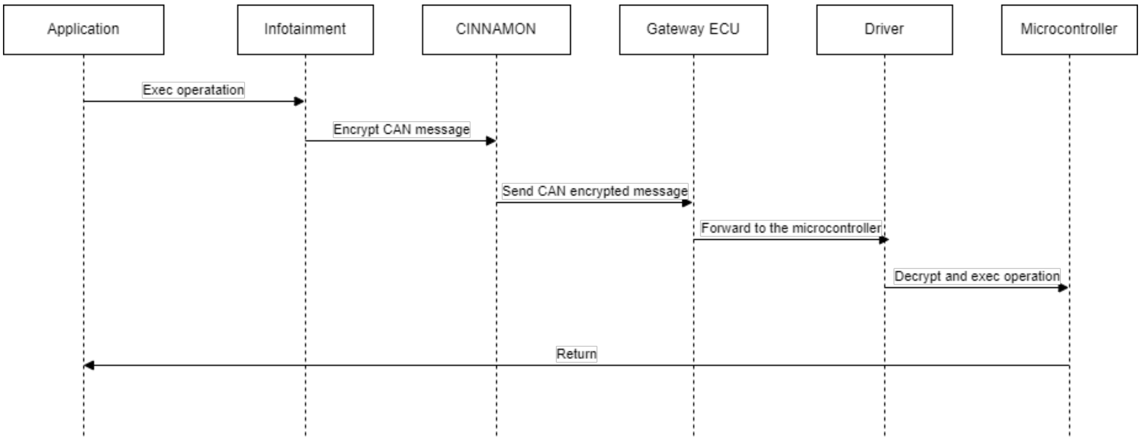


Figure 2.7: Stack application through sequence diagram

boards in our setup.

The laptop may also send manually crafted frames so that we can check broadcast reception, namely by both boards in our testbed.

2.6.2 Implementation complexities

Our initial choice to use Chaskey as a MAC function upon the basis of its specifications was a lucky one. The function was reasonably easy to implement and appreciably fast since the initial experiments. Tags were truncated to 24 bits.

However, the encryption algorithm had to be chosen with care. Our obvious, initial candidate was AES but it produced a data field of at least 128 bits, while we aimed at a data field of 64 bits only so that it could be accommodated in just one frame. On the other hand, a 64 bit version of AES would be weaker and is not standardised. We also experimented with DES, 3DES and Blowfish, but their main drawback for our application was the computational overhead. By contrast, SPECK64/128 [10] uses a 128 bit key, produces a 64 bit output and is lightweight, so it turns out the optimal candidate here.

Further complexity derived from the implementation of the FVM, which, of course, behaves differently from all other boards. We had to decide whether and how to simulate it or whether to program it specifically as with the other boards. While our original impulse was towards the first route, we soon found a convenient way to take the second, as we shall see.

2.6.3 Implementation choices

In light of what we just discussed, our implementations adopt SPECK64/128 [10] to encrypt and decrypt CAN frames, Chaskey to calculate MACs and the two FV mechanisms introduced above at specification level.

A CINNAMON secured CAN frame is formed by reducing the dimension of the payload. Then, a freshness value is used to guarantee that the frame content is fresh. To complete the data field, an additional block is used for the Message Authentication Code (MAC), which ensures authentication and integrity. Finally, the entire 64 bits of the payload are encrypted to ensure confidentiality. Therefore, our implementations of CINNAMON and its current profiles on CAN bus take the MAC-then-Encrypt approach, as specified in Figure 2.8. In operational terms, the sender ECU extracts a key, Key_m, from the CSM and uses it to generate the MAC of payload and FVT; it then truncates the MAC as MACT, extracts another key, Key_e from the CSM to encrypt payload, FVT and MACT. The receiver ECU extracts its copy of the key Key_e (SPECK implements symmetric cryptography) and uses it to decrypt the data field; it then selects payload and FVT, extracts its copy of the key Key_m to re-compute their MAC, hence it truncates that MAC and checks its correspondence with the received MACT.

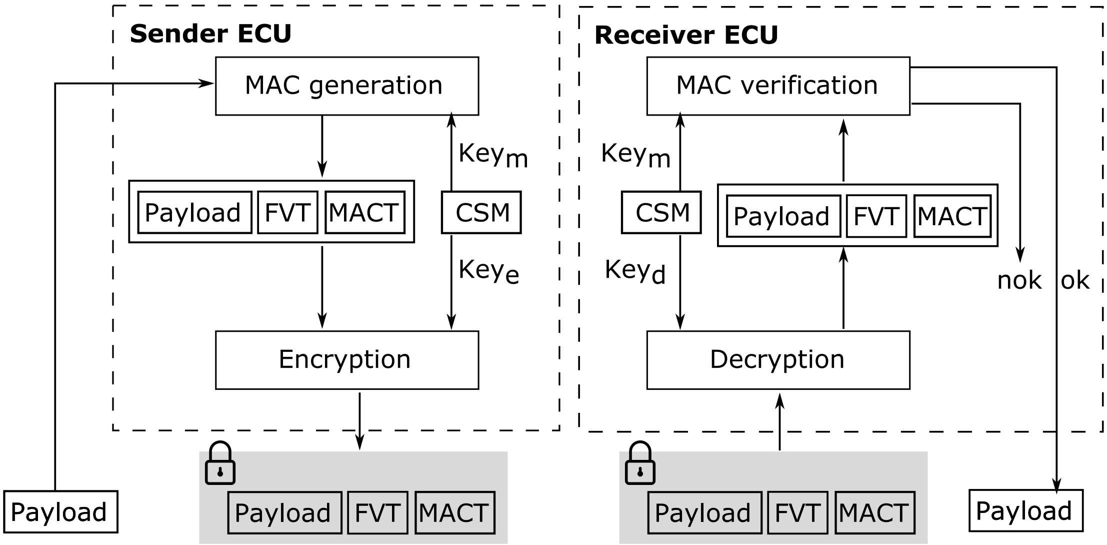


Figure 2.8: Complying with CINNAMON on CAN bus

We are aware that, in general, depending on the chosen algorithm and on the length of the frames, the MAC-then-encrypt approach may turn out less secure than the encrypt-then-MAC approach due to message padding, which may allow an attacker to break the security of the message rebuilding [126]. However, this risk is zeroed in our case because there is no padding needed due to the fixed length of the considered messages.

There is a second reason in support of our choice. The MAC-then-encrypt approach encrypts 64-bit long frames (using encryption algorithms with 64-bit block size and no need for padding). By contrast, using the encrypt-then-MAC approach according to the AUTOSAR specification, the payload (or payload plus FVT) is shorter than 64 bits hence padding would be needed. Most importantly, in frames where the 64 bits are already taken, adding a MAC would necessarily require the transmission of an additional frame to contain it [42].

Another effective choice made through our implementations is about the FVM.

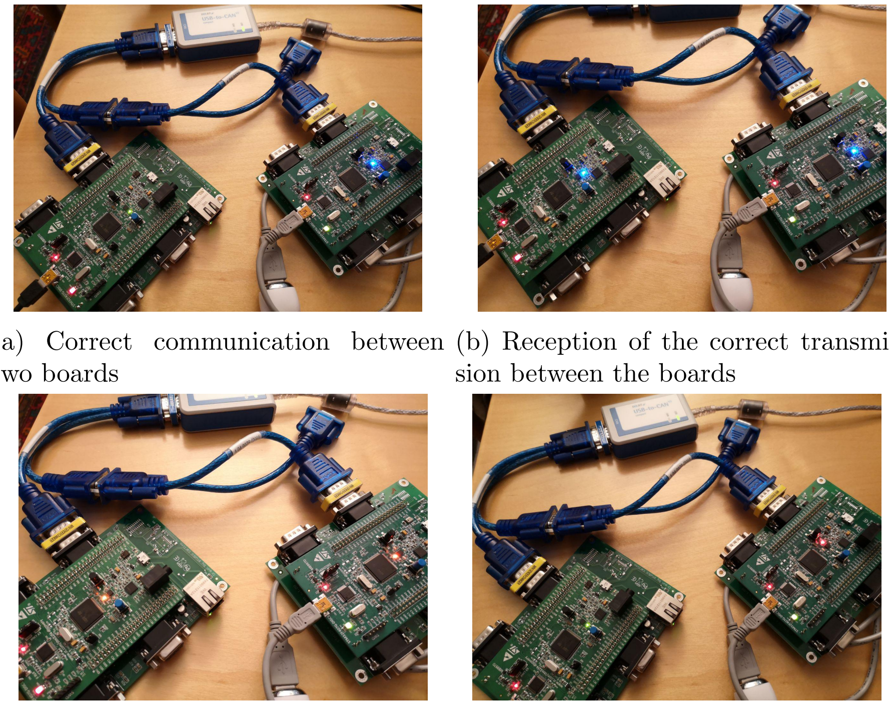
We opt to simulate it through IXXAT canAnalyser 3 [75], which works well off-the-shelf on our laptop to forward CAN frames back and forth between the two boards. However, it does not provide adequate management of the FC in Single Counter style, namely for Security Profile 2. In the homologous SecOC profile, AUTOSAR adopts a centralised management, precisely by the FVM, of the FC for all frames. On one hand, it is convenient to just use canAnalyser 3 as is but, still, the management of the FC has to be implemented from scratch. So, we take a *distributed state matrix* approach to enable each ECU to keep track of the FC value associated with each frame ID. Therefore, each ECU implements a state matrix to store the FC currently associated to each of the frame IDs the ECU can handle.

2.6.4 Visual Cues

Once CINNAMON is deployed on our boards, we can observe the visual cues from its leds to get a confirmation of the operations that the boards performed. In particular, Figure 2.9 shows four notable scenarios. The red led can be noticed on the left hand side of all boards, near the Mini B power connector, indicating operation.

- Figure 2.9a may refer to all security profiles. The left board sent a frame successfully hence turned on the green led. The right board receives that frame successfully hence turns on the blue led.
- Figure 2.9b may refer to all security profiles. The laptop sent a frame, both boards received it successfully hence turn on their respective blue led.
- Figure 2.9c may only refer to Security Profile 2 and Security Profile 3. Both boards received updated FVs from the FVM hence turn on their respective orange led.

- Figure 2.9d may only refer to Security Profile 2 and Security Profile 3. Both boards were originally synchronised, then the left board was reset and, after that, sent a frame successfully hence turned on the green led. The right board is no longer synchronised because the left board was reset, hence the right board discards the received frame and turns on the red led.



(a) Correct communication between two boards

(b) Reception of the correct transmission between the boards

(c) Correct broadcast reception of new freshness values

(d) Synchronisation failure between two boards

Figure 2.9: Led colours representing various board states

These scenarios will be recalled below to demonstrate our implementations of CINNAMON Security profiles.

2.6.5 Security Profile 1 Implementation

Our implementation of CINNAMON Security Profile 1 on CAN bus reviews the CAN frame structures as depicted in Figure 2.10, with 40 bits for the payload and 24 bits for the MAC. These fields are encrypted coherently with the profile specification and our implementation choices seen above.

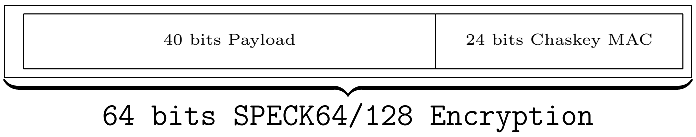


Figure 2.10: Data field of a Security Profile 1 frame

Once the code is deployed on the ECUs, the experiment starts when we press the blue button on the sending board to trigger the preparation of a CINNAMON secured CAN frame. This computes the MAC and performs encryption, building a new frame with a data field as seen in Figure 2.10. The board then sends the frame on the CAN bus to its peer.

The receiver board performs the reverse operations. It decrypts the frame, uses the 40 bits of the payload to calculate the MAC, truncates it and compares it to the version stored in the last 24 bits of the data field. The outcomes of this experiment can be seen in Figure 2.9a.

Another experiment is necessary to check that both boards can receive a frame at the same time. We compute a data field through MAC and encryption, as required by Security Profile 1, and assign it to a frame ID accepted by both boards. Then, we manually input such a frame to canAnalyser 3 and both boards receive it successfully hence turn on the blue led. The outcomes of this experiment can be seen in Figure 2.9b.

Section 2.7.1 outlines the relevant code snippets.

2.6.6 Security Profile 2 Implementation

We have seen that CINNAMON Security Profile 2 extends Security Profile 1 by integrating a FV into the secured frames.

To comply with this on CAN, we implement a modification to the structure of the frame by reducing the payload space as shown in Figure 2.11. Thus, the frame contains a payload of 32 bits, a FV of 8 bits based on Single Counter and a MAC of 24 bits. Encryption remains the final step prior to sending the frame.

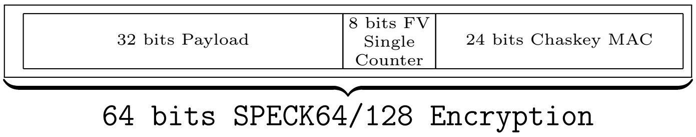


Figure 2.11: Data field of a Security Profile 2 frame

We conventionally use frame ID 102 to carry the FV sent by the FVM. As noted above, each of our ECUs implements a state matrix to store the FC currently associated to each of the frame IDs the ECU can handle. For example, let us refer to our ECUs as #10 and #40 and analyse how they synchronise though the FC and the participation of the FVM in our implementation.

When a vehicle is powered on, the FVM generates a random value that represents the freshness value FV. Then, the FVM communicates FV to all ECUs, which are all assumed to be able to receive frames from the FVM. When an ECU wants to generate a frame with a certain frame ID, it verifies that the FV received by the FVM is greater than the FC the ECU itself stores for that frame ID. If so, then the ECU initialises the FC with the FV provided by the FVM. Otherwise, it increases the FC for that frame ID by one and sends it as FV.

According to the structure of the DBC file, each ECU can handle a specific set of frame IDs and, in turn, each frame ID can be associated with a set of signals. Table 2.10 provides an example.

Table 2.10: Example DBC fragment

ECU No.	Frame Identifier	Signal
#10	201	EngineSpeed
		OilLevel
		FuelConsumption
		...
	205	RHHighBeamFail
		LHHighBeamFail
		...
	...	...
#40	205	RHHighBeamFail
		LHHighBeamFail
		...
	305	RainSensor
		LowFuelWarning
		...
	...	...
...	...	...

Because each ECU stores the last value of the FC that was set for a frame ID, the state matrix of an ECU may look, at some point, as the one in Table 2.11.

Table 2.11: Example state matrix for ECU #10

ECU #10	
Frame Identifier	Freshness Counter
201	15
205	7
...	...

When an ECU wants to send payload on a specific frame with a certain ID, it queries its state matrix to get the FC value associated with that ID. Then, the ECU increases the retrieved FC by one and assigns it as the new FV for the frame to be sent. At the same time, the ECU updates by one also the FC stored in its state matrix for that frame ID. For example, considering the state matrix of Table 2.11, when ECU #10 wants to send the frame whose ID is 201, it finds its associated FC of 15, assigns 16 as FV and updates the stored FC as 16.

On the receiving side, the frame freshness is checked by using the converse mechanism. For example, let us consider the state matrix illustrated in Table 2.12.

Table 2.12: Example state matrix for ECU #40

ECU #40	
Identifier	Freshness Counter
201	15
305	3
...	...

To validate the FV that comes with a received frame, the receiving ECU compares it to the incremented (by one) version of the FC stored in its state matrix for the ID of the received frame. If they differ, then the ECU rejects the frame, otherwise it accepts it and stores the updated FC in its table. For example, if ECU #40 receives a frame with ID 201 and FV of 16, and if its state matrix is as in Table 2.12, then the ECU accepts the frame and updates as 16 the FC stored for the frame. The outcomes of this experiment can be seen, once more, in Figure 2.9a.

If an ECU receives a frame from the FVM, it follows a different procedure. It overwrites its FC values as the frame dictates. This aims at improving synchronicity, and in fact the FVM sents out its frames periodically. For example, if ECU #40 receives a frame from the FVM dictating that the current FV for frame ID 201 is 20, then the ECU updates as 20 the value 15 its example state matrix seen in Table 2.12. The outcomes of this experiment can be seen in Figure 2.9c.

Section 2.7.2 outlines the relevant code snippets.

2.6.7 Security Profile 3 Implementation

This Security Profile manages the FV using the multiple freshness counter approach, so there is a sender-side function to generate the FV and a receiver-side function to reconstruct and verify the FV. Therefore, the data field resembles that of the previous Security Profile with the only difference on the FV counter, as shown in Figure 2.12.

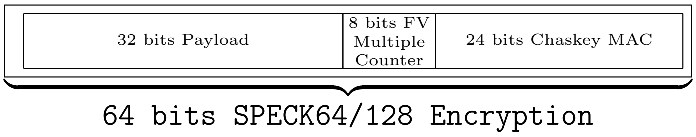


Figure 2.12: Data field of a Security Profile 3 frame

The generation and validation algorithms of FV based on Multiple Counter take into account four counters: *Trip Counter*, *Reset Counter*, *Message Counter* and *Reset Flag*, as Figure 2.13 shows.

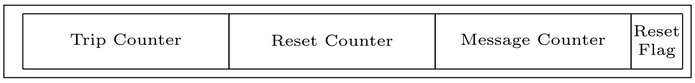


Figure 2.13: Structure of counters in multiple freshness counter approach

These counters are increased according to specific conditions provided by AUTOSAR [9]. In particular, the Trip Counter is increased by one when the FVM starts, resets and when the power status changes. The Reset Counter is increased by one at regular time intervals. The Message Counter is increased by one value for each message transmission. Instead, the Reset Flag is represented by the two least significant bits of the Reset Counter. The FVM maintains only the Trip Counter and the Reset Counter that are initialized to one. Instead, the ECU slaves use all four counters which are initialized to zero.

The FV generation algorithm performs a comparison between the latest and previously sent values of Trip Counter and Reset Counter respectively. If the comparison returns a positive result, then the FV is composed of the previously sent values of the Trip Counter and the Reset Counter, while the Message Counter will be equal to previously sent value plus one and the Reset Flag will be equal to the value from the lower end of the Reset Counter to the previously sent value.

Upon frame reception, freshness verification works as follows. The FVV is generated by performing various comparisons on the four counters. All such comparisons may produce 15 different relative conditions among the counters [9], and the value of the receiver's FV is specifically constructed in each case. For example, in condition termed "Format 1" [9], the algorithm checks whether the latest value of the Reset Flag is equal to the received value, whether the latest value of the Trip Counter equals the previously received value and whether the previously received value of the Message Counter is smaller than the received value. If all these succeed, then the FV will be composed by the previously received values of Trip Counter, Reset Counter and Message Counter Upper, which is the range not included in the truncated version. A more extensive treatment can be found in Appendix 2.4.2.

We implemented the extraction of the various counters from the FV Multiple counter by means of a function that transforms hexadecimal into an array containing its binary representation. For example if we have the hexadecimal value 87, our method will execute the `BinToHex` function and build the array 1|0|0|0|0|1|1|1.

To verify that the complex synchronicity demanded in this security profile worked, we successfully run all four experiments whose visual cues are in Figure 2.9. In particular, we also cross-checked that, by resetting the sender board hence breaking synchronicity, the receiver board would discard the frame. The outcomes of the latter experiment can be seen in Figure 2.9d.

Section 2.7.3 outlines the relevant code snippets.

2.7 Code Snippets

2.7.1 Security Profile 1

Code 2.1 shows a snippet of our implementation for sending frames according to Security Profile 1. More specifically, the sender executes the first `memcpy` which takes as input respectively: the `resMAC` variable which will contain the MAC result, the `chas_mac_create` function which takes the frame as input and returns the MAC and finally the third parameter concerns the size of the `resMAC` variable. Next, the second memcpy is performed which is necessary for creating the frame encryption. In fact, the first parameter of the memcpy is `result_Enc` which is the variable that will contain the whole encrypted frame, the second parameter is the `speck_Enc` function which takes the frame as input and calculates its encryption and the third parameter concerns the size of the `result_Enc` variable. Finally, the `CAN_send` function is executed which manages the transmission of the frame on the bus and takes as input the CAN controller, the frame and a timeout. If the function `CAN_send` is successful, then the green LED will light up, otherwise the red one will light up. .

Code 2.1: Code snippet from sender in Security Profile 1

```
memcpy(resMAC, chas_mac_create(frame), sizeof(resMAC));
memcpy(result_Enc, speck_Enc(frame), sizeof(result_Enc));
if(CAN_send(1, &result_Enc, 0x0F00) == CAN_OK){
  GPIOD->BSRRH = 0xF000;
  GPIOD->BSRRL = 0x1000;
}else{
  GPIOD->BSRRH = 0xF000;
  GPIOD->BSRRL = 0x2000;
}
```

Code 2.2 shows a snippet of our implementation for receiving frames according to Security Profile 1. In more detail, the first line executes the `memcpy` function which takes as input the `result` variable which will contain the decoded frame, the `speck_Dec` function which will perform the decoding and the last parameter concerns the size of the result variable. Subsequently, the calculation and comparison of the MAC is carried out. Finally, if the MAC check gives a positive result, the blue LED will light up, otherwise the red one will light up.

Code 2.2: Code snippet from receiver in Security Profile 1

```
memcpy(result, speck_Dec(msg.data), sizeof(result));
if(chas_mac_can(result)){
    GPIOD->BSRRH = 0xF000;
    GPIOD->BSRRL = 0x8000;
}else{
    GPIOD->BSRRH = 0xF000;
    GPIOD->BSRRL = 0x4000;
}
```

2.7.2 Security Profile 2

Code 2.3 shows a snippet of our implementation for sending frames according to Security Profile 2. The first row shows that operations are triggered upon pressing the onboard button. Then, the board increases by one the FV associated with a given ID and assigns it to a specific position of variable `tmp_frame`, the array used for all preliminary operations needed to create the secured frame. Then, the board sets an example payload and executes the `chas_MAC_create` function to calculate the corresponding MAC. It truncates the MAC and concatenates it to the payload. After that, the board invokes the cryptographic function `speck_Enc` thereby encrypting the 64 bit payload. Finally, it can be seen that the `CAN_send` API sends the frame just built and the green LED is turned on to signal that all operations have been successfully completed.

Code 2.3: Code snippet from sender in Security Profile 2

```
if (isPressed() == 0x01){
 for(i=0; is<stateMatrixLength; i++){
  if (freshMatrix[i][0] == 0x602){
   freshMatrix[i][1] = freshMatrix[i][1]+1;
   tmp_frame[4] = freshMatrix[i][1]; //Freshness Value
   break;
  }
 }
 tmp_frame[0]=0x20; tmp_frame[1]=0x21; tmp_frame[2]=0x22;
     tmp_frame[3]=0x23; //example and temporary payload for a MAC
 memcpy(resMAC,chas_MAC_create(tmp_frame),
 sizeof(resMAC)); //resMAC has the chaskey MAC tag
 for (h=0; h<8;h++){ //concatenation of the payload with the truncated MAC tag
  frame[h] = tmp_frame[h];
   if (h == 5){
    frame[5]=resMAC[0];
    frame[6]=resMAC[1];
    frame[7]=resMAC[2];
    break;
   }
  }
 memcpy(result_Enc,speck_Enc(frame),sizeof(result_Enc)); //Encryption of the frame with SPECK
 for (h=0; h<8;h++) //result of the encryption copied to the data structure for transmission
 can.data[h] = result_Enc[h];
 if(CAN_send(1, &can, 0x0F00) == CAN_OK){
   GPIOD->BSRRH = 0xF000;
   GPIOD->BSRRL = 0x1000;
 }else{
   GPIOD->BSRRH = 0xF000;
   GPIOD->BSRRL = 0x2000;
 }
}
```

Code 2.4 shows a snippet of our implementation for receiving frames according to Security Profile 2. Upon reception of a frame, the board decrypts it and stores its contents in the frame variable. Then, the board verifies the MAC using the `chas_MAC_check` function, which returns 1 if verification succeeds, 0 otherwise. The board now searches the frame ID in its own State Matrix. If the ID exists, then the board increases by one the corresponding FV in the State Matrix, otherwise the board ignores the frame. There follows a comparison between the FV just updated in the State Matrix and the FV received within the frame. If they correspond, then the board switches on the blue led, otherwise it turns on the red led to indicate lack of synchronisation on FV and discards the frame.

Code 2.4: Code snippet from receiver in Security Profile 2

```
memcpy(frame, speck_Dec(can.data), sizeof(frame)); //the "frame" variable contains the decrypted frame
if(chas_MAC_check(frame)){ //function that verifies the MAC, returns 1 if the operation was successful
 for(i=0; i<stateMatrixLength; i++){
  if (stateMatrix[i][0] == can.id && stateMatrix[i][1]+1 == frame[4]){ //check the FV
   stateMatrix[i][1] = stateMatrix[i][1]+1;
   GPIOD->BSRRL = 0x8000; // turn on Blue LED
   break;
  }else
    GPIOD->BSRRL = 0x4000; // turn on Red LED
 }
}
```

2.7.3 Security Profile 3

Code 2.5 shows a snippet of our implementation for sending frames according to Security Profile 3. When a new frame is transmitted, the sender ECU executes the `genFreshSender()` function. It takes four variables as inputs: the last *Trip Counter* value received by the FVM, the last *Reset Counter* value received by the FVM, the values sent previously of the *Trip Counter* and *Reset Counter*. The function checks whether the Trip Counter and Reset Counter values received by the FVM are the same as those previously sent. Based on this check, the FV generation algorithm is performed. The resulting FV consists of the Trip Counter, Reset Counter, Message Counter and Reset Flag. In addition, the aforementioned function also generates the FVT value, which is represented by the last less significant bits of the Message Counter and the Reset Flag. The FVT value will be subsequently inserted in the fifth byte of the frame and will be used by the receiver to build again the FV. Once it is generated, the module builds a frame made up of four bytes of payload and four bytes of FV. At this point, the function to calculate the MAC is invoked. Subsequently, we build a secure frame consisting of four bytes for the payload, the fifth byte is used for the FVT and the remaining three bytes for the truncated MAC. Finally, the entire frame is encrypted with SPECK64/128 and sent into the bus.

Code 2.5: Code snippet from generation of the FV multiple counter

```
if( latestTrip == PrevTrip && latestRst == PrevRst ){
       TripCntSender = PrevTrip;
       RstCntSender = PrevRst;
       MsgCntSender = MsgCntSender+1;
       for (i = 7; i >= 0; i--){
           aBitResetFlag[index]=MID(RstCntSender,i,i+1);
           aBitMsgCnt[index]=MID(MsgCntSender,i,i+1);
           index++;
       }
       ResetFlagSender = aBitResetFlag[6]*10 +aBitResetFlag[7];
       FVTrunk = ((aBitMsgCnt[6]*10 +aBitMsgCnt[7])*100)+ResetFlagSender;
       FVTrunk = BinToHex(FVTrunk);
       ResetFlagSender = BinToHex(ResetFlagSender);
}else{
       TripCntSender = latestTrip;
       RstCntSender = latestRst;
       MsgCntSender = 0x01;
       for (i = 7; i >= 0; i--){
           aBitResetFlag[index]=MID(RstCntSender,i,i+1);
           aBitMsgCnt[index]=MID(MsgCntSender,i,i+1);
           index++;
  }
  ResetFlagSender = aBitResetFlag[6]*10 +aBitResetFlag[7];
  FVTrunk = ((aBitMsgCnt[6]*10 +aBitMsgCnt[7])*100)+ResetFlagSender;
  FVTrunk = BinToHex(FVTrunk);
  ResetFlagSender = BinToHex(ResetFlagSender);
}
```

Code 2.6 shows a snippet of our implementation for verification frames according to Security Profile 3. Upon receiving the frame, the ECU performs the decryption operation and obtains the frame containing the payload, FVT and MAC. The receiver invokes the `genFreshReceiver()` function that takes as input the Trip

Counter and Reset Counter parameters received by the FVM and Trip Counter and Reset Counter previously received by the FVM. In addition, the function takes as input the FVT value contained in the fifth byte of the received frame. Then, a data structure is built to maintain the payload and the FV given as output by the aforementioned function. The MAC is calculated on the data structure to carry out the verification. If the check ends successfully, then the ECU will turn on a blue LED to indicate a positive result, otherwise it will turn on the red LED.

Code 2.6: Code snippet from verification of the FV multiple counter

```
if (latestvalRFlag == rcvRFlag){
        //Format 1 2
        if (LatestTripCntFVM == PrevTrip && LatestRstCntFVM == PrevRst){
        for (i = 7; i >= 0; i--){
            aBitPrevMsgReceiver[index]=MID(PrevMsgCnt,i,i+1);
      index++;
      }
      index = 0;
      PrevRcvUpper = (aBitPrevMsgReceiver[4]*10 +aBitPrevMsgReceiver[5]);
      PrevRcvUpper = BinToHex(PrevRcvUpper);
   PrevRcvlower = (aBitPrevMsgReceiver[6]*10 +aBitPrevMsgReceiver[7]);
   PrevRcvlower = BinToHex(PrevRcvlower);
   rcvLower = (aBitFVTrunk[4]*10 +aBitFVTrunk[5]);
   rcvLower = BinToHex(rcvLower);
   //format 1
   if(PrevRcvlower < rcvLower){
       TripCntReceiver = PrevTrip;
       RstCntReceiver = PrevRst;
```

```
    MsgCntReceiver = PrevRcvUpper*100+rcvLower;
    MsgCntReceiver = BinToHex(MsgCntReceiver);
    ResetFlagReceiver = (aBitFVTrunk[6]*10 +aBitFVTrunk[7]);
    ResetFlagReceiver = BinToHex(ResetFlagReceiver);
   }
   else if(PrevRcvlower >= rcvLower){
       TripCntReceiver = PrevTrip;
       RstCntReceiver = PrevRst;
       MsgCntReceiver = PrevMsgCnt+0x01+rcvLower;
       MsgCntReceiver = BinToHex(MsgCntReceiver);
       ResetFlagReceiver = (aBitFVTrunk[6]*10 +aBitFVTrunk[7]);
       ResetFlagReceiver = BinToHex(ResetFlagReceiver);
      }
     }
     //Format 3
     else if (LatestTripCntFVM >= PrevTrip && LatestRstCntFVM >=
         PrevRst){
         TripCntReceiver = LatestTripCntFVM;
         RstCntReceiver = LatestRstCntFVM;
         MsgCntReceiver = 0x00+(aBitFVTrunk[4]*10 +aBitFVTrunk[5]);
         MsgCntReceiver = BinToHex(MsgCntReceiver);
         ResetFlagReceiver = (aBitFVTrunk[6]*10 +aBitFVTrunk[7]);
         ResetFlagReceiver = BinToHex(ResetFlagReceiver);
}
```

2.8 Runtimes

We measured the runtimes of our prototype implementations of CINNAMON and report them here. It seems fair to conclude that the additional computations that our module induces only slightly affect the overall performances, despite the fact that our code is only a proof-of-concept and our testbed only contains inexpensive hardware.

Table 2.13 shows the runtimes of each algorithm in microseconds on a board at 168 MHz. Not surprisingly, encryption and decryption take longer than the other operations but the excess seems acceptable.

Table 2.13: Runtimes per primitive

Algorithm	Time [μs]
Chaskey (MAC)	0.43
SPECK 64/128 (Enc/Dec)	5.36
FV Gen/Ver Single Counter	0.02
FV Gen/Ver Multiple Counter	0.04

It is also interesting to assess the total runtimes of sending or receiving operations for each security profile. We found out that both operations take the same runtimes for each profile, a non-surprising finding due to the fact that the computational overhead is the same in both cases. Then, Table 2.14 shows the total runtimes of sending or receiving operations, including frame generation upon sending or frame validation upon receiving.

Table 2.14: Runtimes per profile

Profile	Time [μs]
CINNAMON Security Profile 1 (SPECK + Chaskey)	5.79
CINNAMON Security Profile 2 (SPECK + Chaskey + FV Single Counter)	5.81
CINNAMON Security Profile 3 (SPECK + Chaskey + FV Multiple Counter)	5.83

These runtimes remain unvaried over subsequent executions. It can be concluded that CINNAMON adds less than 6μs to generate or validate a secured frame in any of its security profiles, a finding that we deem very promising for applications demanding a secure CAN bus.

2.9 Related Work

In this section, we discuss novelties and advantages of CINNAMON with respect to the state of the art. The discussion identifies and revolves around the following six features $\mathrm{F}_1 \ldots \mathrm{F}_6$, which are relevant to the industrial uptake of secure CAN communication:

F1. Standard CAN. This feature holds of a protocol when all fields of the frame, which the protocol defines, conform to size and contents as specified by the CAN standard [74].

F2. Frame rate equal to CAN's. This is true for a protocol that does not need to increase the CAN's frame rate.

F3. Payload size not smaller than CAN's. This holds of a protocol that preserves the standard CAN size of 64 bits for the payload size.

F4. Standard AUTOSAR. This holds of a protocol that conforms to the prescriptions of the latest AUTOSAR standard [9]. Note that profiles were introduced in the AUTOSAR standard only in 2014.

F5. No ECU hardware upgrade. This holds of a protocol when it requires no upgrade to the ECUs that can run the CAN protocol, hence no additional features or computational power are needed for the units.

F6. No infrastructure upgrade. This is similar to the previous feature but concerns the network and the overall infrastructure that supports the protocol. Therefore, it is true for a protocol that executes on the same network that underlies the CAN, without additional, dedicated nodes.

Since the current version of SecOC module has been introduced in 2014 and slightly revised and improved till 2021, the existing solutions to secure the CAN bus can be naturally divided into ante 2014 and post 2014.

Among the ante 2014 solutions there are:

- CANAuth [124], 2011. It is based on CAN+ [150], which is an extension of the basic CAN protocol in which the data rate is extended in such a way that more bytes can be sent (up to 16 CAN+ bytes for each CAN byte) in the same frame. The drawback of this protocol is that it requires to change the transceivers, which must be more powerful to manage the CAN+ data rate.

This implies that using CANAuth has an impact on hardware, which must be upgraded.

- LCAP [68], 2012, aimed to guarantee message authentication, resistance to replay attacks, and backward compatibility at the same time. It is based on some out-of-band protocol like CAN+. The main drawback is the use of broadcast-based authentication, which increases the traffic in a way directly proportional to the number of nodes in the network.

- MaCAN [99], 2012, is a centralised authentication protocol based on broadcast-based authentication, so it requires CAN+ or CAN FD. However, the same protocol was found to be flawed [30].

- Libra-CAN by Groza *et al.* [66], 2012, a protocol based on a MAC calculated using MD5. Its main drawbacks are high bandwidth and the introduction of hardware capable of understanding and manage the new frame format: Libra-CAN protocol is based on CAN+ instead of on CAN.

Referring to the AUTOSAR standard, all solutions listed above do not appear to be based on the requirements and guidelines described in the SecOC module requirements and specification. Moreover, most of them require to redesign the vehicle network architecture to or introduce new nodes or upgrade the ECUs to manage new protocols.

The solutions proposed after 2014 are the following ones:

- TACAN [145] shares a master key between an ECU and the *Monitor Node* to generate shared session keys. These are assumed to be stored in a Trusted Platform Module (TPM) [72]. Each ECU embeds unique authentication frames into CAN frames and continuously transmits them through covert channels, which can be received and verified by the Monitor Node. TACAN aims at mitigating suspension, injection and masquerade attack. TACAN does not address confidentiality.

- CaCAN [81] introduces a key distribution phase. Hence, the protocol needs a new component in the architecture to act as a monitoring node. Frames are not sent in broadcast but on a peer-to-peer basis. The protocol is simulated but not implemented on micro-controllers.
- LeiA [105] uses MAC to authenticate messages: for each message, the protocol sends a message in plaintext and another one with the MAC of the message. LeiA rests on a 29-bit message identifier, which is coherent with CAN 2.0B [74].
- CANcrypt [3], 2017, is closely related to our work but does not follow AUTOSAR guidelines. Also TLS-based approaches are valid but demand extra-vehicular Internet connectivity and are limited to time-critical applications due to performance overhead [146].

All these protocols present pros and cons. Table 2.15 represents a contrastive analysis of the main entries in the related work with respect to all six features. Notably, no protocol ticks all features, but LeiA and CINNAMON are the only ones that are both CAN compliant, based on the AUTOSAR guidelines and, at the same time, require no upgrade to each ECU, or network augmentation with additional components. CANcrypt does not strictly follow all AUTOSAR profiles. In fact, it does not implement any freshness values algorithm able to mitigate the replay attack. Moreover, LeiA and CINNAMON have alternate features F2 and

F3. While LeiA keeps the CAN payload size of 64 bits, it doubles each frame, a feature that may produce some safety concerns, as discussed elsewhere [41]. On the contrary, both CINNAMON and CANCrypt satisfy F2 but not F3. Both protocols rely on the CAN frame size of 64bit. The main difference is that CANcrypt is designed to be applied to only a subset of messages in a quite small network due to the high introduced overhead. Contrarily to most of the other protocols, such as CaCAN, we implement CINNAMON on micro-controller boards resembling the real behaviour and computational power of ECU. The obtained performances are very promising because the new modules does not introduce additional overhead on the communication bus.

Moreover, recent work analyses the impact of introducing security over functional properties of vehicles. Dariz *et al.* [40, 41] presented a trade-off analysis between security and safety when a security solution based on encryption is applied on CAN messages. The analysis is presented considering different attacker models, packet fragmentation issues and the residual probability of error of the combined scheme. Also Groza *et al.* [65] and Stabili *et al.* [115] targeted the delicate relation between security and safety. Also, a framework for the specification and automatic generation of security features for communications among AUTOSAR-compliant components must be mentioned [22]. It allows AUTOSAR designers to add security specifications to the communication model through a dedicated software tool. However, it has not yet been practically used to advance new components or protocols that would combine confidentiality with authentication and integrity.

Table 2.15: Contrastive analysis of CINNAMON with reference to the related work

	CANAuth [124]	MaCAN [99]	LCAP [68]	Libra-CAN [66]	TACAN [145]	CaCAN [81]	LeiA [105]	CANcrypt [3]	CINNAMON [17, 18]
F1. Standard CAN	✗	✗	✓	✗	✓	✓	✓	✓	✓
F2. Frame rate equal to CAN's.	✗	✗	✗	✗	✓	✗	✗	✓	✓
F3. Payload size not smaller than CAN's.	✗	✗	✗	✗	✓	✗	✓	✗	✗
F4. Standard AUTOSAR	✗	✗	✗	✗	✗	✗	✓	✗	✓
F5. No ECU hardware upgrade	✗	✗	✓	✗	✗	✓	✓	✓	✓
F6. No infrastructure upgrade	✓	✗	✓	✓	✗	✗	✓	✓	✓
	1	0	3	1	3	2	5	4	5

2.10 Discussion

In this Chapter it has been argued that confidentiality is an essential property also in the automotive sector, and that the use of the technical security measure to obtain it, encryption, is already pervasive and also in line with the current European Regulation on the processing of data. Hence, such a use of cryptography should be extended to the automotive domain, and the recent Addendum 154 to UN Regulation 155 emphasises this [83].

Therefore, this Chapter presented CINNAMON [17, 18], an AUTOSAR based basic software module aims at confidentiality, integrity and authentication combined together and, at the same time, enhanced with the freshness attribute. In particular, the introduction of confidentiality makes reverse engineering operations of an attacker harder. For example, understanding communications would require brute-forcing cipher-texts, a daunting task that would, in turn, require the gathering of several pieces of information, such as encryption key, encryption algorithm and general frame semantics. It is reassuring that, because all these activities are unlikely to succeed, the attacker will be unable to forge valid frames and practically abuse real cars the way the news have reported so far.

Clearly, CINNAMON can not only contribute from a cybersecurity point of view, but also, through data encryption, reduce automotive data mining and thus also safeguard drivers' privacy. This could also reduce attacks on driver authentication and driver behaviour.

CINNAMON is general and achieves a TRL 4 prototype implementation on CAN bus that complies with it. The observed runtimes are promising and only negligibly increase those of SecOC, which does not use encryption. This result supports the claim that securing in-vehicle communications cryptographically is currently possible hence viable for large-scale deployment.

Table 2.6: Construction of Freshness Value for Reception [9]

Construction Format	Condition			Construction of freshness value for verification			
	(1) Reset Flag comparison	(2) Trip Counter and Reset Counter comparison	(3) Message Counter (lower end) comparison	Trip Counter	Reset Counter	Message Counter (Upper)	Message Counter (Lower)
Format 1	Latest value = Received value	Latest value = Previously received value	Previously received value < Received value (no carry)	Previously Received value	Previously Received value	Previously Received value	Received value
Format 2			Previously received value >= Received value (with carry)	Previously Received value	Previously Received value	Previously received value+1	Received value
Format 3		Latest value > Previously received value	-	Latest value	Latest value	0	Received value
Format 1	Latest value-1 = Received value	Latest value-1 = Previously received value	Previously received value < Received value (no carry)	Previously Received value	Previously Received value	Previously Received value	Received value
Format 2			Previously received value >= Received value (with carry)	Previously Received value	Previously Received value	Previously received value+1	Received value
Format 3		Latest value-1 > Previously received value	-	Latest value	Latest value-1	0	Received value
Format 1	Latest value+1 = Received value	Latest value+1 = Previously received value	Previously received value < Received value (no carry)	Previously Received value	Previously Received value	Previously Received value	Received value
Format 2			Previously received value >= Received value (with carry)	Previously Received value	Previously Received value	Previously received value+1	Received value
Format 3		Latest value+1 > Previously received value	-	Latest value	Latest value+1	0	Received value
Format 1	Latest value-2 = Received value	Latest value-2 = Previously received value	Previously received value < Received value (no carry)	Previously Received value	Previously Received value	Previously Received value	Received value
Format 2			Previously received value >= Received value (with carry)	Previously Received value	Previously Received value	Previously received value+1	Received value
Format 3		Latest value-2 > Previously received value	-	Latest value	Latest value-2	0	Received value
Format 1	Latest value+2 = Received value	Latest value+2 = Previously received value	Previously received value < Received value (no carry)	Previously Received value	Previously Received value	Previously Received value	Received value
Format 2			Previously received value >= Received value (with carry)	Previously Received value	Previously Received value	Previously Received value+1	Received value
Format 3		Latest value+2 > Previously received value	-	Latest value	Latest value+2	0	Received value

Chapter 3

Drivers' privacy concerns and trust perceptions

Modern cars are evolving in many ways. Technologies such as infotainment systems and companion mobile applications collect a variety of personal data from drivers to enhance the user experience. This chapter investigates the extent to which car drivers understand the implications for their privacy, including that car manufacturers must treat that data in compliance with the relevant regulations. It does so by distilling out drivers' concerns on privacy and relating them to their perceptions of trust on car cyber-security. A questionnaire is designed for such purposes to collect answers from a set of 1101 participants, so that the results are statistically relevant. In short, privacy concerns are modest, perhaps because there still is insufficient general awareness on the personal data that are involved, both for in-vehicle treatment and for transmission over the Internet. Trust perceptions on cyber-security are modest too (lower than those on car safety), a surprising contradiction to our research hypothesis that privacy concerns and trust perceptions on car cyber-security are opponent. We interpret this as a clear demand for information and awareness-building campaigns for car drivers, as well as for technical cyber-security and privacy measures that are truly considerate of the human factor.

The Chapter is structured as follows: Section 3.1 outlines useful insights into personal data in modern cars. Section 3.2 defines the research method used, in particular the questionnaire design, the crowdsourcing task and the statistical approach. Section 3.3 discusses the results obtained from this work. Section 3.4 addresses the difference between the pilot study and the full study. Section 3.5 describes the related works. Section 3.6 summarises and discusses all the contents of the chapter.

3.1 Personal data in modern cars

The cars people are driving at present are complicated cyber-physical systems involving tight interaction between rapidly evolving car technologies and their human users, the drivers. To meet the needs and preferences of (at least) drivers, the infotainment system is more and more integrated with the setups for passengers' physical preferences, such as seating configuration, driving style and air conditioning, as well as for non-physical preferences, such as music to play, preferred numbers to call and on-line payment details.

A plethora of data originates, whose processing enhances the driving experience and exceeds that towards increased support for autonomous driving, a goal of large interest at present. Modern cars also come with Internet connectivity ensuring, at least, that car software always gets over-the-air updates from the manufacturer. Cars expose services remotely via dedicated apps that the driver installs on their smartphone to remotely operate functions such as electric doors, air conditioning, headlights, horn and even start/stop the engine. Therefore, car and driver's smartphone apps form a combined system that exposes innovative services, including locating the car remotely via GPS or even geo-fencing it, so that the app user would be notified if their car ever exceeds a predefined geographical area [4]. Because cars are progressively resembling computers, offering services while treating personal data, they also attract various malicious aims.

The field of car cyber-security shows that software vulnerabilities can be exploited on a Jeep [123], on a General Motors [76] as well as on a Tesla Model S [37]. Such vulnerabilities may, in particular, impact data protection, and the sequel of this manuscript will discuss the variety of personal data treated through cars, thus calling for compliance, at least in the EU, with the General Data Protection Regulation (GDPR) [49]. Car cyber-security is certain to be more modern than car safety, hence our overarching research goal is to understand whether the former is understood as well as the former is. We formulate the hypothesis that privacy concerns decrease when trust perceptions on the underlying security and data protection measures are correspondingly high. For example, it means that if a driver feels that their personal data is protected, then that is because the driver trusts that the car is secure. To assess such hypothesis, this paper does not take a common attack-then-fix approach but, rather, addresses the following research questions pivoted on drivers' perceptions.

RQ1. Are drivers adequately concerned about the privacy risks associated with how that their car and its manufacturer treat their personal data?

RQ2. Do drivers adequately perceive the trustworthiness of their car, in terms of security especially?

We are aware that these research questions are not conclusive, and we have gathered data to specialise the answers by categories of drivers, e.g. by age or education. To the best of our knowledge, this is the first large-scale study targeting and relating privacy concerns and trust perception of car drivers. We took the approach of questionnaire development and survey execution through a crowdsourcing platform. Our goal was to get at least 1037 sets of responses in order for the results to be statistically relevant, as explained below. We first piloted the questionnaire with 88 friends and colleagues with the aim of getting feedback but no significant tuning was required. After crowdsourcing, a total number of 1101 worldwide participants was reached.

We analysed the results obtained from the questionnaire through standard statistical analysis by Pearson's correlation coefficient, Spearman's rank correlation coefficient and Coefficient Phi. In a nutshell, most drivers believe that it is unnecessary for their car to collect their personal data because they find the collection unnecessary to the full functioning of modern cars; this indicates that privacy concerns are low, which in turn may be due to wrong preconceptions, given that cars do collect personal data. Also, it appears that most drivers do not fully agree that their data is protected using appropriate security measures; this may be interpreted as a somewhat low trust on security. To our surprise, pairing these two abstracted results clearly disproves our hypothesis.

3.2 Research method

We took the approach of questionnaire development and survey execution to assess drivers' privacy concerns and perceptions of trust. Specifically, the questionnaire begins with questions regarding the demographics of the participants in order to extrapolate whether perceptions of privacy and trust depend on the demographics of the subjects in our sample. Then the questionnaire continues with 10 questions focused mainly on the topics of privacy and trust on drivers. Finally, we administered it through a crowdsourcing platform and performed a statistical analysis of the responses. Opinions were measured using a standard 7-point Likert scale. With a very low margin of error, of just 4%, and a very high confidence level, of 99%, the necessary sample size to represent the worldwide population is 1037. Our total respondents were 1101, including piloting over 88, so our results are statistically relevant of the entire world — a limitation is that, while Prolific ensures that respondents are somehow geographically dispersed, it cannot guarantee that they are truly randomly sampled from the entire world.

3.2.1 Correlation Coefficients

Correlation coefficients allows to establish whether or not there is a relationship between the data. In the following sections the correlation coefficients used for this analysis are briefly described, each with its own formula, the range of the coefficient and how to interpret it.

Pearson correlation coefficient

Pearson's linear correlation coefficient allows to evaluate a possible linearity relationship between two sets of data [136].

Given n pairs of numerical variables $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, the Pearson correlation coefficient is defined as the covariance between the two sets of data divided by the product of their standard deviation:

$$r_{xy} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

where $\bar{x}$ and $\bar{y}$ represent the arithmetic mean of X and Y dataset respectively.

The coefficient has a value between -1 and $+1$, both included. The stronger the relationship between the two sets of data, the higher is the absolute value of the correlation coefficient. The sign of the coefficient indicates the direction of the correlation. If one variable increases when the second one increases, then there is a positive correlation. In this case the correlation coefficient will be closer to 1. If one variable decreases when the other variable increases, then there is a negative correlation and the correlation coefficient will be closer to -1. A perfect correlation value ($+1$ or -1) means that the correlation between the data sets is perfectly described by a linear function. Correlation coefficient values are accompanied by a significance level (the *p-value*) to establish the reliability of the calculated value. The p-value is a number between 0 and 1 representing the probability that the result would have been obtained if the data had not been correlated. If the p-value is less than 0.01 then the relationship found is statistically significant [135].

Spearman's rank correlation coefficient

The Spearman's rank correlation coefficient is denoted by the Greek letter ρ and it measures the correlation between two numerical variables. To calculate this coefficient the two statistical variables must be sortable since Spearman's correlation coefficient is defined as Pearson's correlation coefficient applied to the ranks. Instead of using the precise values of the variables, the data are ranked in order of size, and calculations are based on the differences between the ranks of corresponding values X and Y [141].

$$\rho_{XY} = r_{rg_X,rg_Y} = \frac{cov(rg_X, rg_Y)}{\sigma_{rg_X}\sigma_{rg_Y}}$$

Thanks to the relationship between this coefficient and the previous one, its interpretation is similar to Pearson's correlation coefficient however, unlike Pearson's coefficient, Spearman's coefficient measures how well a relationship between two variables can be described by any monotonic function. This makes the Spearman coefficient more reliable than the Pearson coefficient in situations where there is a correlation between two sets of data but it is not described by a linear function.

Point-Biserial Correlation Coefficient

The point-biserial correlation coefficient is a correlation coefficient that is used when one of the two statistical variables assumes at most two values [139]. In this case the goal is to determine if the value of the binary variable has any effect on the value of the other numeric variable. To calculate this coefficient the pairs are divided into two separate groups according to the value of the binary variable.

Given n pairs $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ where X is the binary set and Y is the numeric one, we build a group A with the pairs having $x = 1$ and a group B with the pairs having $x = 0$. The point-biserial correlation coefficient between this two datasets is defined as:

$$r_{pb} = \frac{M_1 - M_0}{\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(y_i - \bar{y})^2}}\sqrt{\frac{n_1 n_0}{n(n-1)}}$$

where:

- M_1 is the mean of Y values belonging to group A
- M_0 is the mean of Y values belonging to group B
- n_1 is the number of items in group A
- n_0 is the number of items in group B
- $\bar{y}$ is the mean of the whole Y series

The interpretation of this coefficient is similar to the interpretation of the Pearson coefficient. A high value with positive sign indicates a positive correlation between the binary variable and the numeric variable, i.e. the higher values of the numeric variable tend to be present when the binary variable has value 1. A high value with negative sign indicates a negative correlation: the lower values of the numeric variable tend to be present when the binary variable has value 1. A value close to 0 indicates no correlation.

Coefficient Phi

The Phi coefficient (or mean square contingency coefficient) is denoted with the Greek letter ϕ and it is is a measure of association for two binary variables [31]. This coefficient is calculated from the frequency distributions of the pairs.

Give n pairs $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ with X and Y both binary variables, we count the occurrences as shown of each pair and we define the variables:

- a as the number of $(1, 1)$ occurrences
- b as the number of $(1, 0)$ occurrences
- c as the number of $(0, 1)$ occurrences

- d as the number of $(0, 0)$ occurrences

Once these values are obtained the coefficient is calculated with the following formula:

$$\phi = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}}$$

This measure is similar to the Pearson correlation coefficient in its interpretation. In fact, a Pearson correlation coefficient estimated for two binary variables will return the Phi coefficient.

3.2.2 Engineering the questionnaire

The questionnaire is structured around a preliminary part followed by a core part. The preliminary part begins by asking participants' demographic information, that is, age, gender, nationality, level of education, employment status and job sector. This part is extended by asking participants how many hours a week they spend driving their vehicle. That number may offer somewhat objective information on drivers' knowledge about modern cars — if we assume that driving time correlates with understanding how a car works.

The core part of the questionnaire consists of ten questions, with the first half (questions Q1-Q5) targeting privacy concerns and the second half (questions Q6-Q10) oriented at trust perceptions.

Each half begins with a pair of questions setting up baseline information on respondents, then continues with the essential questions.

Question Q1 seeks participants' subjective evaluation of their own knowledge about modern cars. Then, question Q2 asks respondents whether or not they agree that modern vehicles are similar and comparable to modern computers. This pair of questions derives the respondents' essential understanding of the subject matter, and it is relevant to correlate such information with the remaining questions in the first half.

Question Q3 starts delving into the data processing aspects. It asks participants to select the kind of data they think modern cars collect, with options including personal data, public data, financial data, special categories of data and, ultimately, no data at all. Clearly, such options are inspired to GDPR terminology.

Then question Q4 asks participants whether or not they agree that personal data collected by a modern car about its driver is necessary for the full functioning of the car. This question allows us to understand if drivers believe that, in order to be able to use all the features provided by their vehicle without any sort of limitation, they need to provide their personal data. Question Q5 asks whether or not respondents think it is necessary to transmit the personal data collected over the Internet. In relation to the previous question, participants may agree to provide their personal data to obtain additional features — for instance, statistics about driving style and vehicle usage — as long as their data remain limited to their vehicle.

Question Q6 asks participants whether or not they agree that a modern vehicle safeguards its driver's life. This question introduces the survey part about the trust that drivers pose in their car. In relation to this question, question Q7 asks respondents whether or not they agree that a modern car protects its driver's personal data better than it safeguards its driver's life. Then with question Q8 participants are asked if they agree that a modern car processes the personal data it collects about its driver in a legitimate way that is consistent with the relevant regulations (e.g. the European General Data Protection Regulation also known by its acronym *GDPR* [49] which is effective from May 2018). Question Q9 asks participants whether they agree that a modern car carries out a systematic and extensive evaluation of the personal data it collects about its driver on the basis of automated processing in order to evaluate personal aspects. In fact, according to the current legislation (art. 22 of the GDPR) these processes must be properly declared and explicit consent is required for the proposed purposes. In addition, (art. 32 of) the GDPR requires the use of adequate security measures to protect the rights and freedoms of data subjects. The last question (Q10) prompts respondents to decide whether they agree that the personal data a modern car collects about its driver is protected by suitable technology when the car transmits data over the Internet.

3.2.3 The core part of the questionnaire

The complete list of main questions in the questionnaire follows those already present in [15], but in more detail all questions are listed below.

Q1 Are you knowledgeable about modern cars?

Not at all □ – □ – □ – □ – □ – □ – □ Very knowledgeable about modern cars

Q2 How much do you agree with the following statement: a modern car is similar to a modern computer.

Strongly disagree □ – □ – □ – □ – □ – □ – □ Strongly agree

Q3 What kind of data do you think a modern car collects about its driver?

- □ No data at all
- □ Public data not about the driver
- □ Public data about the driver
- □ Personal data about the driver (e.g. name, address, etc.)
- □ Special categories of personal data about the driver (e.g. racial or ethnic origin, political opinions, religious or philosophical beliefs, trade union membership, genetic data, biometric data, data concerning health or data concerning sex life or sexual orientation)
- □ Financial data about the driver (e.g. credit card number)

Q4 How much do you agree with the following statement: "personal data collected by a modern car about its driver is necessary for the full functioning of the car".

Strongly disagree □ – □ – □ – □ – □ – □ – □ Strongly agree

Q5 How much do you agree with the following statement: "it is necessary that the personal data collected by a modern car about its driver be transmitted over the Internet".

Strongly disagree □ – □ – □ – □ – □ – □ – □ Strongly agree

Q6 How much do you agree with the following statement: "a modern car safeguards the life of its driver".

Strongly disagree □ – □ – □ – □ – □ – □ – □ Strongly agree

Q7 How much do you agree with the following statement: "a modern car protects its driver's personal data better than it safeguards its driver's life".

Strongly disagree □ – □ – □ – □ – □ – □ – □ Strongly agree

Q8 How much do you agree with the following statement: "a modern car processes the personal data it collects about its driver in a legitimate (i.e. coherently with pertinent regulations) way".

Strongly disagree □ – □ – □ – □ – □ – □ – □ Strongly agree

Q9 How much do you agree with the following statement: "a modern car carries out a systematic and extensive evaluation of the personal data it collects about its driver on the basis of automated processing, including profiling (e.g. to evaluate certain personal aspects of the driver to analyse or predict aspects concerning the driver's performance at work, economic situation, health, personal preferences, interests, reliability, behaviour, location or movements)".

Strongly disagree □ – □ – □ – □ – □ – □ – □ Strongly agree

Q10 How much do you agree with the following statement: "the personal data a modern car collects about its driver is protected by suitable technology when the car transmits it somewhere on the Internet".

Strongly disagree □ – □ – □ – □ – □ – □ – □ Strongly agree

3.3 Results

This Section presents the results gathered from all 1101 respondents. The answers are catalogued and statistically studied by analysing indexes of central tendency and correlation coefficients. The indexes of central tendency (mean and median) synthesise with a single numerical value the values assumed by the data. The mean value is coupled with the standard deviation in order to measure the amount of variation of the values. We recall that driving at least 3 hours a week was a prerequisite to enter the study, along with being over 18.

To simplify the analysis of the answers to the core questions, we follow the standard practice of grouping the 7 levels of agreement into three categories. Specifically, if the participants reply with “Strongly agree”, “Agree” or “Somewhat agree”, then we consider their value as “Agreeing”; if instead they select “Neither agree nor disagree”, then we consider them in the category “Undecided”; and finally, if the participants select “Somewhat disagree”, “Disagree” or “Strongly disagree”, then we consider those answers as “Disagreeing”.

3.3.1 Knowledge on modern cars

Question Q1 evaluates the driver's knowledge on modern cars. Considering the values of the mean and the median, shown in Table 3.1, it can be stated that the interviewed sample considers itself knowledgeable about modern cars. The data show that 55% of participants are quite confident about their knowledge, while a minority of the participants (about 29%) think they are not. Finally, 16% of participants think they have average knowledge about modern cars. Thus, considering the answers of the preliminary question, there does not seem to be a substantial difference between those who drive a few hours a week and those who drive more with regard to the level of knowledge they claim to have on modern cars.

Then, question Q2 asks respondents whether or not they agree that modern cars are similar to modern computers. Also this question receives a high rate of agreement. We note that 72% of participants agree that a modern car is similar to a modern computer. Furthermore, it turns out that 14% of them are undecided while 14% of them disagree with the statement. The mean and the median are shown in Table 3.1.

Table 3.1: Q1, Q2 answers and their statistics

Knowledge level	[%]
Knowledgeable about modern cars	55
Average knowledge	16
Not knowledgeable about modern cars	29
Mean	4.37
Median	5
Standard Deviation	1.55

Agreement level	[%]
Agreeing	72
Disagreeing	14
Undecided	14
Mean	5
Median	5
Standard Deviation	1.35

3.3.2 Concerns on data privacy

The first of these questions (Q3) asks participants to select all the categories of data they think a car collects. It must be remarked that this answer allows for multiple choices, so a respondents can choose from multiple categories of data. Table 3.2 shows the answers selected by the respondents. The predominant categories according to the interviewed sample are: "personal data about the driver" (selected by 56% of the sample); "public data about the driver" (selected by 54% of the sample);

"public data not about the driver" (selected by 47% of the sample). A few participants think that their vehicle collects more sensitive data belonging to the special categories of personal (13%) and financial data (11%). Finally, we note that just 8% of the participants think that modern cars do not collect any data at all.

Overall, these results confirm a modest level of awareness in terms of what data a car collects. In particular, while it is positive that the majority (56%) understands that driver's personal data are involved, it is concerning that a similar subset (54%) deem such data about the driver to have been made public. It would be surprising if any car manufacturer's privacy policy stated that the driver's collected data would be made public (and such policies are well worthy of a dedicated comparative study). This potential confusion calls for awareness campaigns, more readability of official documents and innovative technologies to ensure policies are understood. By contrast, a positive sign that a small kernel of participants is highly informed is the appreciable understanding that special categories of personal data (13%) or financial data (11%) may be gathered.

Table 3.2: Q3 answers

Collected data	[%]
Personal data about the driver	56
Public data about the driver	54
Public data not about the driver	47
Special categories of personal data about the driver	13
Financial data about the driver	11
No data at all	8

Question Q4 asks participants whether they think it is necessary to collect personal data to achieve full vehicle functionality. The indexes and a summary of the answers are shown in Table 3.3. It shows that 27% of the participants agree with the statement above, moreover, 19% of them are undecided and 54% of them disagree with the statement. Thus, we could argue that the participants disagree with the statement proposed in the question.

This result can be interpreted in various ways. On one hand, it denounces a false preconception because that the customised, driver-tailored experience that is getting more and more common at present is certain to stand on a trail of data collected about the driver's. It clearly also signifies that drivers are neither adequately informed on what data is being collected and for what purposes, contradicting art. 5 of GDPR, nor have they been able to grant an informed consent, contradicting art. 7 of GDPR.

Table 3.3: Q4, Q5 answers and their statistics

Agreement level	[%]	**Agreement level**	[%]
Agreeing	27	Agreeing	21
Disagreeing	54	Disagreeing	65
Undecided	19	Undecided	14
Mean	3.35	Mean	2.97
Median	3	Median	3
Standard Deviation	1.58	Standard Deviation	1.67

Moving on to the answers of question Q5, it can be noticed that just 21% of the sample agrees to the transmission of data over the Internet, only 14% of participants are undecided moreover 65% of them disagree with the statement. This means that the sample is not very convinced to send personal data over the Internet. Table 3.3 shows agreement levels and indexes of Q5's answers.

This may again be interpreted as a wrong preconception because it is clear that remote services, including eCall, location-tailored weather forecasts, music streaming and many others, must generate Internet traffic.

3.3.3 Perceptions of trust on safety

Question Q6 asks whether participants agree that a modern vehicle safeguards the life of its driver. The agreement levels and the indexes of central tendency are shown in Table 3.4. It turns out that 77% of participants agree with the statement above, then just 8% disagree with the statement, and 15% of them are undecided.

Question Q7 asks participants whether a modern car protects its driver's personal data better than its driver's life. It appears that a part of the sample is undecided with this statement (26%), just 18% of participants agree with the statement moreover 56% of them disagree. Table 3.4 shows also that the indexes of central tendency are not as high when compared to the previous question.

There is considerable uncertainty in front of this question, if not for the majority's expression of disagreement (56%). It signifies that trust on security still has a great lot to grow in comparison to trust on safety, perhaps due to the much longer establishment of the latter. It is well known that trust may take a long time to root, and car security is certain to be a somewhat recent problem.

Table 3.4: Q6, Q7 answers and their statistics

Agreement level	[%]
Agreeing	77
Disagreeing	8
Undecided	15
Mean	5.26
Median	5
Standard Deviation	1.20

Agreement level	[%]
Agreeing	18
Disagreeing	56
Undecided	26
Mean	3.26
Median	4
Standard Deviation	1.46

3.3.4 Perceptions of trust on security

Question Q8 asks whether the data collected from the vehicle is legitimately processed according to the relevant regulations. Table 3.5 shows that 44% of the participants agree with this statement moreover 25% disagree and the rest of them (31%) are undecided.

Trust one the legitimacy of the data processing is not higher than 44%. This indicates, once more, that car drivers need to be better informed, first of all. Conversely, this means that the majority, 56%, are not sure about the legitimacy of the processing of their personal data. Being informed correctly is essential for raising awareness, which in turn is essential for trust building.

Question Q9 asks if participants believe that the personal data collected is systematically analysed and evaluated using automated processes (including profiling). From Table 3.5, around 42% of participants agree with this statement, moreover 32% of them disagree and 26% are undecided with the statement.

This question is designed to be self-contained and understandable by everyone. A notable 42% show concern that profiling takes place, which may be taken to signify a correspondingly low trust on the security of the treatment. There is no official public information on whether car manufacturers really carry out profiling but, if this were the case, then a Data Protection Impact Assessment, pursuant art. 35 of GDPR, would have been necessary.

The last question (Q10) asks whether the participants feel that the data transmitted over the Internet are protected by adequate technologies. Table 3.5 confirms the representation that agrees with the question (46%) to be considerable. The fact that those who agree do not exceed the majority of the sample clearly indicate, also in this case, room for improving drivers' trust on security.

Table 3.5: Q8, Q9, Q10 answers and their statistics

Agreement level	[%]
Agreeing	44
Disagreeing	25
Undecided	31
Mean	4.28
Median	4
Standard Deviation	1.31

Agreement level	[%]
Agreeing	42
Disagreeing	32
Undecided	26
Mean	4.07
Median	4
Standard Deviation	1.43

Agreement level	[%]
Agreeing	46
Disagreeing	32
Undecided	22
Mean	4.19
Median	4
Standard Deviation	1.49

3.3.5 Correlations

There are statistically significant correlations that arise by analysing the relevant coefficients over the data obtained from the sample. In brief, Pearson's linear correlation coefficient, denoted by the letter r, allows us to evaluate a possible linearity relationship between two sets of data [137]. Spearman's rank correlation coefficient, denoted by the Greek letter ρ, measures the correlation between two numerical variables; these must be sortable because Spearman's correlation coefficient is defined as Pearson's correlation coefficient applied to the ranks [140]. The Phi coefficient (or mean square contingency coefficient), denoted with the Greek letter ϕ, is a measure of association for two binary variables and is calculated from the frequency distributions of the pairs [138]. Correlation coefficient values are accompanied by a significance level (the *p-value*) to establish the reliability of the calculated value. The p-value is a number between 0 and 1 representing the probability that the result would have been obtained if the data had not been correlated. If the p-value is less than 0.01 then the relationship found is statistically significant [134].

The analysis is divided into three sub-sections for clarity. The first looks at the core questions, those from Q1 to Q10, to focus on general correlations between knowledge on the subject matter, privacy concerns, trust perceptions on safety and on security. The second specifically targets the possible interdependencies between data from the core questions and participants' demographics. The third aims at the interdependencies between the data of the main questions and the number of driving hours performed by the participants.

Core question analysis

We noted a significant correlation between question Q1 and question Q2 ($\rho = 0.48$, $p < 0.001$). Therefore, it seems that participants who are knowledgeable about modern cars also think that modern cars are similar to modern computers, reinforcing the conclusion. Moreover, thanks to the correlation between question Q1 and question Q4 ($\rho = 0.35$, $p < 0.001$) we can state that those who consider themselves informed about modern cars also believe that the data collected by the car is necessary for the full functioning of the car. This aligned with our own, specialist view. There is also a significant correlation between question Q1 and question Q6 ($\rho = 0.40$, $p < 0.01$), that is, those who are knowledgeable about modern cars think that a modern car safeguards its driver's life. Somewhat surprisingly, it appears that Q1 does not significantly correlate with later questions on trust on car security, signifying that trust on security must grow even for those who are knowledgeable about the field.

We calculated the Phi coefficients between the answers of question Q3 to determine if there are any associations, i.e. whether there are pairs of categories of personal data that appear together in the answers. The coefficient values are shown in Table 3.6, and it becomes apparent that there are only two values that may establish a possible association. The Phi Coefficient obtained between the couple "Special categories of personal data" and "Financial data about the driver" is 0.3255, which means that those who think that financial data are collected by modern cars also think that special categories of personal data are collected as well. This exhibits a correct preconception because financial data are routinely grouped with special categories of data. Also, given the 0.3363 value, we notice that drivers who think "Special categories of personal data" are collected from the car, also think that "Personal data about the driver" are collected, emphasising a correct understanding that personal data also include the special categories (of personal data).

Table 3.6: Phi Coefficients of question 3

ϕ	No Data	Fin	Spec	Pub	Pub$_{driver}$	Pers
No Data	1					
Fin	-0.0920	1				
Spec	-0.0999	**0.3255**	1			
Pub	-0.2371	0.0004	-0.0624	1		
Pub$_{driver}$	-0.2973	0.1468	0.0759	0.0255	1	
Pers	-0.3136	0.2332	**0.3363**	-0.2099	0.1743	1

Moving on to question Q4, there is a strong statistically significant correlation between question Q4 and question Q5: both the Pearson coefficient and the Spearman coefficient have very high values ($r = 0.55$, $\rho = 0.68$) both with a reliability value $p < 0.001$. In consequence, we can affirm that those who think that it is necessary to collect personal data for the full functioning of their vehicle also think that this data should be transmitted over the Internet. There is also another statistically significant correlation between question Q4 and question Q8 ($r = 0.39$, $\rho = 0.50$, $p < 0.01$ for both), showing that those who agree to the collection of personal data also think that the data are processed legitimately in a manner consistent with the relevant regulations. Both of these can be taken as indications that those with modest privacy concerns show some trust on security, but we are mindful of the generally low agreement with Q4 and Q5 and only fair agreement with Q8 noted above.

Question Q5 correlates only moderately with question Q8 ($\rho = 0.48$, $p < 0.01$) but more significantly with question Q10 ($r = 0.36$, $\rho = 0.52$, $p < 0.01$). It follows that those who think that data should be transmitted over the Internet, also think that this data will be adequately protected during transmission. This shows that trust on security is broad if present.

Spearman's correlation coefficient detects a moderately significant correlation between question Q6 and question Q8 ($\rho = 0.45$, $p < 0.01$), so that it seems that those who think that a modern car safeguards its driver's life also think that the personal data collected are processed legitimately according to the relevant regulations in force. This seems a positive outcome in terms of a spreading of trust on safety over trust on security. It is unfortunate that this correlation is not very strong, and we deem it highly desirable to develop socio-technical security and privacy measures to reinforce it in the future.

There is a statistically significant correlation between question Q7 and question Q10 ($r = 0.38$, $\rho = 0.52$, $p < 0.01$). In fact, those who think that a modern car protects its driver's personal data better than it safeguards its driver's life also think that the personal data are protected by adequate technology when the vehicle transmits it over the Internet. The Spearman's correlation coefficient also shows a significant correlation between question Q7 and question Q8 ($\rho = 0.47$, $p < 0.01$), that is, those who think that a modern car protects its driver's data better than it safeguards its driver's life also think that the personal data collected are processed legitimately according to the relevant regulations. These results confirm that trust on security is somehow "logical" in the sense that it covers all relevant elements.

There is also a significant correlation between question Q8 and question Q9 ($\rho = 0.41$, $p < 0.01$), it appears that those who think that modern cars carry out a systematic and extensive evaluation of personal data also think that their data are processed in a legitimate way according to relevant regulations. This correlation suggests that drivers who consent to the evaluation of their personal data even consent to profiling — perhaps too lightheartedly, raising concern that the potentially negative consequences of profiling may not be fully understood at present. It may be inferred that drivers are not fully aware that it would be their right to object to profiling, as prescribed by art. 22 of GDPR.

We also noted a moderate correlation between question Q9 and question Q4 ($\rho = 0.45$, $p < 0.01$). Those who think that in order to use the full functionality of the car it is necessary to provide personal data also think that this data is analysed and studied according to automatic processes to evaluate personal aspects of drivers. This reconfirms that profiling is somewhat ill-understood. There is also a statistically significant correlation between question Q9 and question Q5 ($\rho = 0.46$, $p < 0.01$) indicating that those who think that their data are analysed by automatic evaluation processes also think that they are transmitted over the Internet. This outcome correctly indicates that potential profiling does not take place aboard the car.

There is a statistically significant correlation between question Q10 and question Q4 ($r = 0.37$, $\rho = 0.49$, $p < 0.01$), that is, those who think that the personal data collected by the vehicle is necessary for the full functioning of the car also think that their data is adequately protected when transmitted over the Internet. Once more, modest privacy concerns lead to some trust on security. Finally, there is a statistically significant correlation between question Q10 and question Q8 ($r = 0.51$, $\rho = 0.64$, $p < 0.01$), so we can argue that those who think that their personal data is processed lawfully also think that the data are adequately protected over the Internet. Here is yet another confirmation that trust on security, if at all present, covers all relevant aspects.

Demographic-based analysis

It is also relevant to analyse the responses to assess whether perceptions of privacy and trust depend on the demographics of the subjects in our sample. We grouped the sample by age, gender, education level and employment status. Thus, we defined two groups by gender, three groups by age (18-24, 25-34 and over 35), three groups by educational level (high school diploma or less, undergraduate degree, graduate degree and Ph.D.), and three groups by employment status (student, employed, non-employed).

Subjects in the 18-24 age group felt less knowledgeable about modern cars, with the percentage of positive responses to question Q1 at 49%, and with the median of responses decreasing to a value of 4. However, the same group of subjects showed more trust in technologies used to protect the transmission of data over the Internet. In fact, the 18-24 group had a 51% agreement rate to question Q10, with the median increasing to 5, while the other two groups had about 40% of agreement. As for the group 25-34, we noticed a stronger disagreement in the answers to question Q4. The percentage of disagreement for these subjects is 61%, higher than the other two groups (51%). This group seems to be more interested in privacy than the other two. We also noted lower percentages of responses to question Q3 from the over-35 group. Specifically, the category "public data about the driver" was selected by 41% of the group, while the category "public data not about the driver" was selected by 38%. This group thinks that less data is collected from the vehicle than the others. Furthermore, the over 35 group disagreed more with the statement in question Q9 than the 25-34 group, the percentages of disagreement being 38% and 29% respectively. Over 35s seem to be the least aware of profiling. Finally, the over-35 group was the group with the lowest number of disagreement to question Q5 (55%) compared to the other two (67%), indicating that this group is less opposed to data transmission over the Internet.

The analysis based on gender showed the following results. Men showed more knowledge and trust in their vehicle. In particular, in question Q1 61% of men responded positively compared to the value of women (43%), the median of women's responses was also lower (4). However, both gender exhibited a similarly high agreement (respectively 72% and 73%) to the claim that modern cars are similar to computers in Question Q2. In question Q3, women selected more "personal data about the driver" than men (62% vs. 53%), but less "public data not about the driver" (38% vs. 52%). Finally, the percentage disagreeing with Q5 was higher among women, 69% compared to 62% among men, so women seem to be a little more concerned about data transmission.

The level of education influenced the responses to question Q1. We noted that higher degrees of education corresponded to higher percentages of positive responses.

In particular, only 48% of the subjects belonging to the group "high school diploma or less" answered positively, while with the group "graduate degree and Ph.D." the percentage rises to 65%. In question Q2, the "graduate degree and Ph.D." group also had higher percentages of agreement than the others (80%). We also noted that "graduate degree and Ph.D." were less wary of the transmission of personal data on the Internet. The percentage of disagreements in the answers to question Q5 are lower (58%) than in the other two groups (68%). The group "high school diploma or less" is less concerned about profiling. Only 36% agreed with the statement proposed in question Q9, while the other two groups agreed more (44% undergraduate degree, 49% graduate degree and Ph.D.). Finally, we noted that the group that trusts the car the least is the undergraduate degree group, with 60% disagreeing in question Q7 compared to 51% for the other groups.

The study based on employment status showed a few significant results. The employed group felt more informed than the other categories (students and non-employed), having a percentage of positive responses to question Q1 of 63%, with a median of responses of 5. Non-employed showed more privacy concerns. They responded more negatively to question Q4, with a percentage of disagreement of 62% compared to 51% of students and 54% of employed. Finally, students were more trusting of data security measures, with 51% agreeing in Q10 compared to 42% for the other categories.

Driving hours analysis

It is also important to analyse how the responses perform with respect to the number of driving hours per week. This set of correlations can be applied to the full study because the number of driven hours was always collected.

We defined four groups for driving hours (3-6, 7-9, 10-12 and over 13 hours a week).

From the data obtained it is very evident that the groups of driving hours 7-9, 10-12 and over 13 feel very informed about modern cars, in fact for these three groups we obtained agreement values that are respectively 68%, 65% and 75%. It therefore appears that the group of drivers who drive more than 13 hours a week felt more informed about modern cars than the other two, this could be due to the fact that those who drive more feel more experienced in the automotive field, in this specific case this category of drivers feel with more knowledge about cars.

On the other hand, the 3-6 hour range are felt less than the other ranges in fact the positive percentage of Q1 demand for this range is 45%. Furthermore, we note that all four groups agree with the fact that modern cars are similar to modern computers, in fact the agreement rate is above 69% for all four groups.

Regarding question Q3 we note that a majority of the group of people who drive more than 13 hours a week think that no data is collected from the cars and in general this group believes that less data is collected from the vehicle than the others.

From these analyses we noted that both groups disagree with questions Q4 and Q5. In fact, the disagreement values for both questions are higher than 50%. We believe these groups are more interested and care about their privacy.

Continuing with the analysis we note that all four groups agree with Q6, in fact the agreement rates are in all above 74%. Finally, the same group of stakeholders showed greater confidence in the technologies used to protect data transmission over the Internet. In fact, all groups showed agreement rate above 41% for Q10.

3.4 Pilot study vs. full study

This Section starts off by detailing the results of our pilot study on 88 participants and continues by highlighting how these varied towards the full results presented above. Table 3.7 indicates that the pilot respondents feel quite well informed about modern vehicles (Q1), and Table 3.9 signifies that they feel the collection is mostly of personal and public data, with no relevant mention of financial information or special categories of personal data (Q3).

The remaining results are summarised in Table 3.8. The majority of participants agrees that systems and technologies present in modern cars are increasingly similar to modern computers (Q2). Regarding the necessity to collect personal data for the full functioning of the car, the participants are to be equally divided between those who agree, those who disagree and those who neither agree nor disagree (Q4). Continuing, we note that participants mostly disagree on the need for a modern car to transmit over the Internet the personal data collected its driver(Q5). By contrast, we see that the vast majority agree that a modern car safeguards the life of its driver (Q6). Continuing, we note that a large part of the sample does not agree with the fact that a modern car protects its driver's personal data better than it safeguards its driver's life (Q7). We note, instead, that half of the participants agree that a modern car processes personal data about its driver in a legitimate way (Q8). The next answers tell us that half of the sample think that their data is analysed and studied by the vehicle systems in order to evaluate some personal aspects (Q9), showing that there are some privacy concerns. The last question shows that the majority of the sample agree that their data is protected using appropriate methods and techniques (Q10), indicating a relevant trust perception.

All material is now available to closely compare the results of the pilot study to

Table 3.7: Answers to the Q1 of the pilot study

	Q1
Knowledgeable about modern cars	70%
Average knowledge	17%
Not knowledgeable about modern cars	13%

Table 3.8: Answers to the pilot study questionnaire in percent

	Q2	**Q4**	**Q5**	**Q6**	**Q7**	**Q8**	**Q9**	**Q10**
Agreeing	83%	32%	26%	80%	22%	50%	52%	51%
Disagreeing	6%	33%	48%	6%	41%	34%	31%	23%
Undecided	11%	35%	26%	14%	37%	16%	17%	26%

those of the full study. The outcomes of the comparison are fully given by numbers in Table 3.10 and graphically represented through the spider charts in Figure 3.1.

From the Table 3.10, it can be seen that the results on both the Q2 and Q6 studies are very similar for the "Agreeing" and "Undecided" categories. This implies that, for these questions, the pilot study and the full study represent the sample in the same way despite the different sample sizes. Furthermore, this similarity of values can be seen through the Figure 3.1, where the points of all the questions are almost overlapping. Therefore, looking at the shape and position of the points in Figure 3.1, it can be seen that for the "Agreeing" index the pilot represented the sample well.

On the other hand, it can be seen that the other answers to the other questions are not so similar, especially with regard to the categories "Disagreeing" and "Undecided". An easy example concerns question Q4 which gave different results (33% for the pilot study and 54% for the full study). Although an eye is drawn to question Q9 whose "Disagreeing" values for the pilot study and the full study are almost identical. This implies that for question Q9 the pilot study represented the sample well. Finally, as regards the "Undecided" category, it can be noted that the

Table 3.9: Answers to the Q3 of the pilot study

	Q3
Personal data about the driver	69
Public data about the driver	56
Public data not about the driver	44
Special categories of personal data about the driver	15
Financial data about the driver	15
No data at all	1

shapes of the graphs are very different and therefore the pilot study does not show a good approximation for the full study.

To support the work done and the graphs obtained, we decided to calculate the Pearson correlation coefficients between the two studies. Recall that if the values are very close to one then the studies are correlated. The results obtained for the categories “Agreeing” and “Disagreeing” are 0.99 and 0.89 respectively, which implies a strong correlation between the two studies, as could also be seen from the spider graphs. On the other hand, the value of Pearson's coefficient for the category “Undecided” is 0.14, which implies an absence of correlation. Therefore, we can say that these correlation values confirm what is shown in the spider graphs, namely a good approximation for the first two categories and less for the last one. In conclusion, we can say that in general the two categories (“Agreeing” and “Disagreeing”) correlate quite well. On the other hand, with regard to the category “Undecided” we can state that the participants of the pilot study showed less uncertainty than the participants of the full study and this generated a low correlation between the two studies.

Table 3.10: Comparison of results between the pilot study and the full study

	Q2		**Q4**		**Q5**		**Q6**		**Q7**		**Q8**		**Q9**		**Q10**	
	Pilot	Full	Pilot	Full	Pilot	Full	Pilot	Full	Pilot	Full	Pilot	Full	Pilot	Full	Pilot	Full
Agreeing	83%	72%	32%	27%	26%	21%	80%	77%	22%	18%	50%	44%	52%	42%	51%	46%
Disagreeing	6%	14%	33%	54%	48%	65%	6%	8%	41%	56%	34%	25%	31%	32%	23%	32%
Undecided	11%	14%	35%	19%	26%	14%	14%	15%	37%	26%	16%	31%	17%	26%	26%	22%

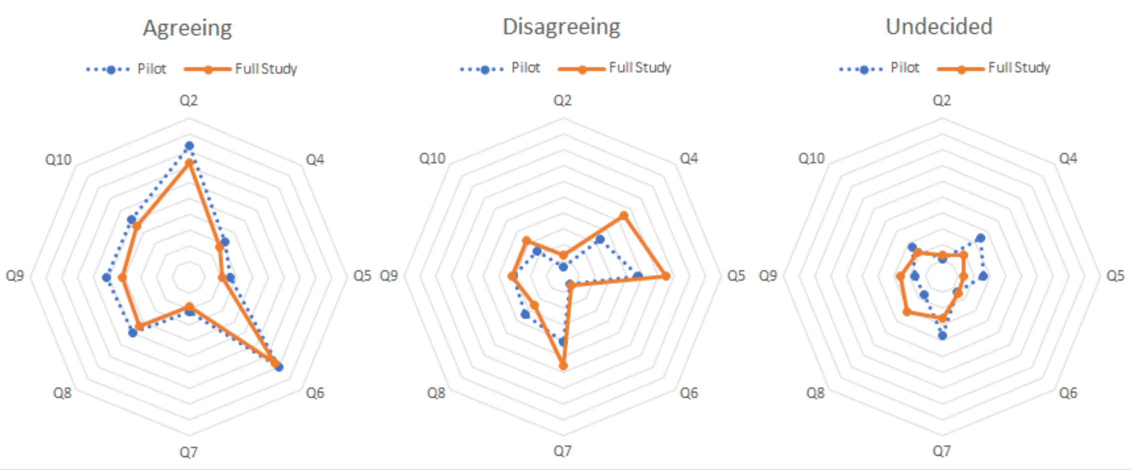


Figure 3.1: Comparison chart between pilot and full study.

3.5 Related Work

In 2014, Schoettle and Sivak [110] surveyed public opinions in Australia, the United States and the United Kingdom regarding connected vehicles. Their research noted that people (drivers as well as non-drivers) expressed a high level of concern about the safety of connected cars, which does not seem surprising on the basis of the novelty of the concept at the time. However, participants demonstrated an overall positive attitude towards connected car technology, with particular interest in device integration and in-vehicle Internet connectivity. In 2016, Derikx et al. [44] investigated whether drivers' privacy concerns can be compensated by offering monetary benefits. They analysed the case of usage-based auto insurance services where the rate is tailored to driving behaviour and measured mileage and found out that drivers were willing to give up their privacy when offered a small financial compensation. Therefore, what appears to be missing is a study on drivers' understanding on the amount and type of personal data that modern cars process, which is the core of this paper.

There are relevant publications on drivers' trust on car safety but are limited to self-driving cars. Notably, Du et al. [96] conducted an experiment to better understand whether explaining the actions of automated vehicles promote general acceptance by the drivers. They found out that the specific point in time when explanations were given was crucial for their effectiveness — explanations provided before the vehicle started were associated with higher trust by the subjects. Similar results were obtained by Petersen et al. [104] in another study in 2019. They manipulated drivers' situational awareness by providing them with different types and details of information. Their analysis showed that situational awareness influenced the level of trust in automated driving systems, allowing drivers to immerse themselves in non-driving activities. Clearly, the more people are aware of something, the more trust they manage to place in it.

It is clear that modern cars technologies are not limited to self-driving features. Modern cars include innumerable digital components, often integrated in the infotainment system, which interact with drivers and collect their data. It follows that modern cars process personal data to some extent, as detailed in the next Section, hence car manufacturers must meet specific sets of requirements to comply with the relevant regulations. Therefore, it becomes important to assess drivers' concerns on their privacy through their use of a car and drivers' trust on the security (also in relation to their trust on the safety) of the car.

3.6 Discussion

In this chapter it can be seen that the study has been carefully designed to clip drivers' privacy concerns and perceptions of trust with the ultimate goal of evaluating the research hypothesis that low privacy concerns imply perceptions of trust. Crowdsourcing was used to gather a representative sample of participants. The responses were then analysed in isolation and statistically correlated, yielding a great many insights. There would be little use in developing amazing technical security and privacy measures for preserving drivers' privacy and the security of their cars in case drivers are not adequately concerned about the privacy issues bound to their driving and yet do not trust the security of their cars at an appropriate level. That case is confirmed by the results of the study, thus contradicting the research hypothesis.

Precisely, one might think that the privacy concerns that have emerged are insufficient in the current technological context. It would have been more positive if drivers had shown greater awareness of the personal data involved in their driving, how the processing of such data is crucial to provide services tailored to the driver, and the fact that this quality of service often requires the transmission of data over the Internet. Unfortunately, the opposite scenario occurs. A somewhat logical explanation for the low privacy concerns could be high trust in security, but surprisingly, trust in security was also quite low. Thus, the only way to read the overall result is that privacy is generally poorly understood by drivers, so we learn that more information needs to be provided to raise awareness and thus form correct privacy concerns and corresponding appropriate trust perceptions. Thus, it can be argued that all this must be the end effect for the development of ever more advanced technical security and privacy measures.

The correlations among answers could be seen as somewhat logical. For example, knowledge on the field correlates with adequate privacy concerns and well-related trust perceptions. It is noteworthy that the potentially negative implications of profiling on the freedoms of natural persons are far from being well received at the moment. Trust on security is much less represented than trust on safety, arguably because the former derives from a less rooted perception in our society due to the relatively young age of the technologies that should support it. Moreover, trust on cyber-security is normally broad, that is, if it is present to some extent, it then covers all relevant aspects. Ultimately, it can be argued that correlations also justify the need for more awareness and trust building campaigns.

The results obtained provide several values and insights. They can be read in support of the ISO/SAE DIS 21434 standard. They also offer a solid baseline to conduct a cyber-security and privacy risk assessment on cars following standard methodologies such as ISO/IEC 27005 [73].

Chapter 4

Assessment of the privacy risks aboard top-selling cars

The general objectives of safety, cyber-security and privacy intertwine aboard modern cars. While safety is always the primary concern, security comes close and intertwines safety, because modern cars are more and more intensively computerised and. For example, no driver would like her cabin preferences to be altered remotely by someone else and, worse still, her lane assistance system to be hijacked.

These are the reasons why cyber-security expertise and related socio-technical measures have been ported, since the beginning of the last decade especially, to the automotive world, with the result that it is routine for mechanics today to also get some “logical” access, beside the traditional physical access, to the car they are repairing. A simple car repair session may even occur solely logically, for example to review and reset an error message that got triggered sporadically.

Cyber-security is known to be a circular process that sees the perennial addition of security measures, which may eventually get broken by upcoming attack techniques calling, in turn, for yet more measures. This implies that some cyber-security risk always exists, and notable works have been advanced to asses that risk specifically in the automotive domain, as noted below. The motivating observation of the present chapter is that comparatively less attention has been paid to the privacy risks in the same domain.

Modern cars acquire a variety of data, ranging from music preferences through payment information to environmental information such as temperature, GPS coordinates and camera streams. Some cars explicitly collect the driver's Personally Identifiable Information (PII), starting with the name, hence the mass of data that a car treats is personal data because it can be easily referred to a natural person. When PII is not treated, it could be inferred with high approximation in various ways, including by querying the Public Vehicle Register, in particular by an attacker with data exfiltration aims.

Despite the tight relationship between cyber-security and privacy, which recognises the role of cyber-security measures to protect personal data, we contend that privacy requires a separate argument from cyber-security, particularly in terms of risk assessment, for various reasons. One is that the existing risk assessment frameworks, recalled below, do not appear to revolve around personal data. Another one is the plethora of personal data that is involved, which may even include sensitive data, for example about the driver's health and religion [21]. Moreover, there is evidence that drivers' understanding of the implications on their privacy deriving from their use of a car is somewhat ill-understood [16]. Privacy concerns rise particularly in Europe, where EU Regulation 2016/679, the "General Data Protection Regulation" (GDPR) addresses privacy as highly as an element of "protection of natural persons" [49]. Here comes the full motivation for the work presented in this chapter.

There are two best-known approaches to conduct risk assessment. One is oriented to assets, as pioneered by ISO 27005 [73] and its ancestors. This approach pursues a discourse that is clearly pivoted around the value of assets. When privacy is the overall objective of the risk assessment, as in our case, all assets are types of personal data. When the domain is automotive, such data is of various types, as outlined above, and refers to the driver (and more sporadically to passengers). The other notable risk assessment approach is oriented to threats and is termed STRIDE [88]. It prescribes a process resting on 6 threat categories and produces insights on how threats and threat categories affect a target system such as a car brand, namely all cars of that brand.

The contribution of this chapter is a double assessment of privacy risks about the most common, on the basis of sales data, car brands. The assessment takes both the ISO 27005 and the STRIDE approaches to develop a framework oriented at privacy aboard the different car brands. The framework is demonstrated in practice to derive the brands whose assets are most at risk by the ISO approach and the brands suffering highest threats by the STRIDE approach. The asset-oriented findings are that the top data breach risks affect Tesla, Volkswagen and Audi, while the threat-oriented outcomes highlight that the top risks affect Mercedes on 4 threat categories and Ford, Tesla and Toyota on 2 threat categories. These findings confirm that, by looking at risk from different angles, the two approaches enrich the insights with a complementarity that was not available before. The overall framework is summarised through the flowchart in Figure 4.1, which is described in the sequel of this article.

This is the first large-scale privacy risk assessment exercise concerning specific representatives of the automotive domain, culminating with a discussion on possible additional measures to mitigate the estimated risks. It must be recalled that risk

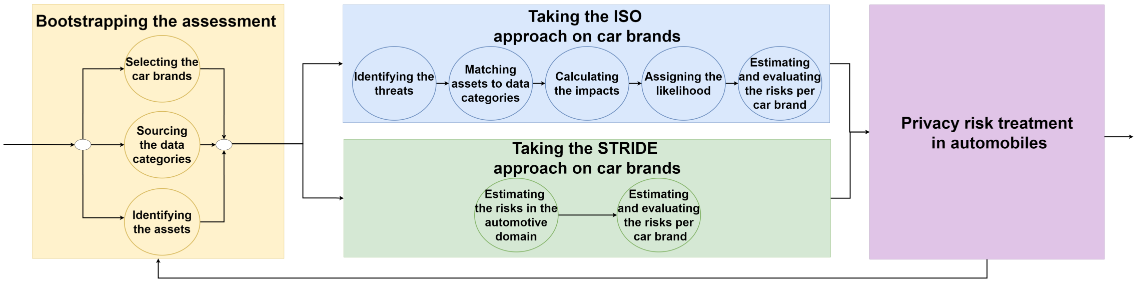

Bootstrapping the assessment
Selecting the car brands
Sourcing the data categories
Identifying the assets
Taking the ISO approach on car brands
Identifying the threats
Matching assets to data categories
Calculating the impacts
Assigning the likelihood
Estimating and evaluating the risks per car brand
Taking the STRIDE approach on car brands
Estimating the risks in the automotive domain
Estimating and evaluating the risks per car brand
Privacy risk treatment in automobiles

Figure 4.1: Flowchart of the framework for the double assessment of privacy risks aboard top-selling cars

assessment is inherently subjective and affected by the assessor's bias, limitations that are typically thwarted by focus groups and by relying on existing evidence in support of the assessment, as shall be seen below. This in turn calls for a need of relevant information, which may be problematic. This chapter sources the relevant information from the web by means of structured queries. Should other sources of information become available, the assessment could be easily reviewed, while our framework would not change.

The Chapter is structured as follows: Section 4.1 and Section 4.2 recall respectively the basics of risk management and risk assessment, and Section 4.3 sets the essentials for a privacy risk assessment in the automotive domain. Section 4.4 demonstrates how to apply the ISO approach on car brands. Section 4.5 complements the presentation with the details of the domain-level STRIDE findings in support of Section 4.6 that demonstrates the STRIDE approach findings on car brands. Section 4.7 introduces possible risk treatment measures, and Section 4.8 comments on the related work. Finally, Section 4.9 summarises and discusses all the contents of the chapter.

4.1 Risk Management

Organisations face internal and external factors and influences that make it uncertain whether they will achieve their objectives. The International Standard ISO 27005 defines the risk as the effect this uncertainty has on an organisation's objectives [73]. Risk management is the process of identifying, analysing and managing the risks that an organisation faces. Risk management can be applied to an entire organisation as well as to specific functions, projects and activities. Once implemented the management of risks enables a company to improve the identification of threats, comply with relevant legal requirements and improve stakeholder confidence and trust. The process of risk management is an iterative process consisting of phases which enable continuous improvement in decision making and performance improvement [47]. Various standards and good practices exist for the establishment of these processes. Organisations, however, tend to create their own instances of these methods in a form most suitable for a given organisational structure, business area or sector. National and international standards are taken as a basis.

Information Security Risk Management is the application of risk management methods to information technology in order to manage IT risks. This, can be a part of an organisation's wider risk management process or can be carried out separately. ISO/IEC 27005 is a set of standards published by the International Organisation for Standardisation (ISO) and the International Electrotechnical Commission (IEC) that provides guidelines and techniques for managing information security risks [73].

This international standard is designed to assist in the implementation of information security and it is part of a larger set of standards in the Information Security Management System (ISMS), the ISO/IEC 27000 series. It is applicable to all types of organisations which intend to manage risks that can compromise the organisation's information security. This standard does not specify any risk management methods, it involves a process that consists of a structured sequence of activities. The information security risk management process consists of context establishment, risk assessment, risk treatment or risk acceptance, risk communication and risk monitoring.

The first phase of the risk management process is context establishment. By establishing the context, the organisation articulates its objectives, defines the external and internal parameters to be taken into account when managing risk and sets the scope and risk criteria for the remaining process. The external context is the external environment in which the organisation seeks to achieve its objectives. It is based on the organisation-wide context but with specific details of legal and regulatory requirements. Internal context, instead, is anything within the organisation that can influence the way an organisation manages risk. During this phase the organisation should also define criteria to be used to evaluate the significance of risk. Some criteria can be imposed by legal and regulatory requirements. Risk criteria factors include how likelihood and impact will be defined, how the level of risk is to be determined and the level at which risk becomes acceptable. Deciding whether risk treatment is required is based on operational, technical, legal, social or environmental criteria or combinations of them.

4.2 Risk Assessment

Risk assessment is a process to describe risks and enable organisations to prioritise risks according to their established criteria. The core sub-processes of risk assessment are risk estimation and risk evaluation. Risk assessment determines the relevant assets, identifies the threats and vulnerabilities that exist or could exist, determines the potential impacts, prioritises the derived risks and ranks them against the risk criteria set in a preliminary context establishment. The overall process is often conducted over several iterations. An overall assessment may be carried out over the risks that generally affect the application domain. Then, using the output of the previous analysis, a further vertical assessment may be carried out over the specific representatives of the domain. This chapter concentrates on the second level of assessment, hence it derives general privacy risk assessment findings about the automotive domain and tailors them to specific car brands.

4.2.1 Asset identification

One of the essential tasks through risk assessment is to create a comprehensive list of assets. An asset is anything that has value to the organisation and therefore requires protection. The definition of assets is not limited to hardware or software. The set of assets includes services, communications, data and infrastructure. The level of detail used through asset identification can be refined in further iterations of the risk assessment. Although each asset needs to be protected, some assets are more critical than others, that is, damage to these assets causes greater damage to the organisation. Each asset is then subject to a valuation process, that is, it is assigned a value determined by the replacement value of the asset and the business consequences of loss or compromise of the asset. These include legal consequences from the disclosure, modification, non-availability and destruction of information. Intuitively, the higher the value, the more important is the asset.

4.2.2 Threat identification

The organisation should identify the general sources of risk, areas of impacts, relevant events and their possible consequences. The aim of this step is to determine what could happen to cause a potential loss and to gain insights into how, where and why the loss might happen. Relevant and up-to-date information is important in identifying risk sources. This is a critical step because a source that is not identified here will not be processed along the subsequent steps. Sources should be considered whether or not they lie under the control of the organisation.

Risk sources help identify threats. While a source of risk is where a risk originates and where it comes from, a threat is any event that may potentially occur from the risk source and would harm assets hence organisations. Threats include both accidental and voluntary events, which may arise from within or from outside the organisation. Some threats may affect more than one asset.

4.2.3 Risk estimation

Risk estimation involves developing an understanding of the risk. It makes it possible to assess the danger of an undesirable event, that is, a threat, in order to define the priority and the urgency of the measures necessary to control the odds that the event occurs.

Risk estimation is commonly qualitative and involves an assignment, typically on a small interval of numbers such as 1 through to 4, on the likelihood that a threat materialises. It also involves an understanding of the impacts that would derive, also in this case to be expressed within some interval of numbers. Impacts normally depend on asset values, as shall be seen in the following. When the estimation is quantitative, the interval numbers are replaced with actual quantities, for example of money or time.

The assessment of threat likelihood is notoriously affected by the human assessor's bias. While some level of subjectivity is unavoidable, bias is routinely thwarted by considering previous pertaining events, such as how often a threat occurred in the past — assuming that the future will not deviate significantly from the past. The likelihood assignment process also depends on how easily a threat can be exploited by skilled and motivated attackers.

A third factor influencing the likelihood assignment comes from the existing controls, precisely from whether they work well against the threats. This is why the existing controls should be identified and their functioning checked. An incorrectly implemented or malfunctioning control could itself be a vulnerability and represent a threat.

Typical likelihood values can be interpreted as follows:

- 1, or *rare/low:* there are valid countermeasures or, alternatively, the motivation for an attacker is very low;

- 2, or *unlikely/medium:* a possible attacker needs to address strong technical difficulties to pose the threat or, alternatively, the efforts are not worth the impacts;

- 3, or *possible/high:* the technical requirements necessary to pose this threat are not high and could be solved without significant effort, furthermore there is a reasonable motivation for an attacker to perform the threat;

- 4, or *likely/very high:* there are no sufficient mechanisms installed to counteract this threat and the motivation for an attacker is quite high.

Risk estimation may take an *asset-oriented* approach or *threat-oriented* approach or both. Asset-oriented estimation revolves around asset values and aims to describe the impacts and their likelihood to produce a risk level. Threat-oriented estimation is somewhat complementary to the previous approach and develops in the opposite direction, being pivoted on threats. Impact and likelihood values may be combined differently, according to various categories of threats as well as to scope and objectives of the overall management process. Taking both approaches in parallel may offer deeper insights, as this chapter demonstrates below.

Independently of the approach taken, the outcome of the assessment is a description of the estimated risks in the form of prose, numbers, colours, or a combination of these. However, these may not be meaningful for the organisation until they undergo evaluation, which is the next step.

The ISO approach

The best-known asset-oriented risk estimation process comes from the ISO 27005, a de-facto standard framework for risk assessment.

The evaluation of impact is based on the typical parameters that characterise the overall objective of the risk assessment process. For example, a privacy objective implies that impact refers to type and volume of personal data as well as to the level of identifiability of data subjects. For example, because Personally Identifiable Information (PII) has a high value, any threats to it get a correspondingly high impact.

If the impact values are given on the same interval as that for the likelihood values, than a popular approach to estimate the risk level is through a *risk matrix*. Each cell of a risk matrix expresses a specific pair of likelihood and impact values.

It must be remarked that ISO 27005 [73] does not prescribe a standard risk matrix, hence organisations may create their own, depending on the specific features of their activities and business sector.

If likelihood or impact values are decimal numbers, they could still be mapped through a risk matrix, but in this case a purely numerical treatment is considered more effective. In consequence, risk can be estimated by a standard formula:

$$\begin{aligned} &Risk(threat, asset) = \\ &\quad Likelihood(threat, asset) \times \\ &\quad Impact(threat, asset) \end{aligned} \tag{4.1}$$

Lifting this formula at the level of the organisation to produce the organisation's privacy risk can be done in various ways, but could get complicated because of the necessary generalisation on both parameters. For example, fixing a threat, its likelihood for the organisation could be derived as the maximum or the average of the threat likelihood on all assets. Such outcomes for all threats could then be combined, by a similar function, as the overall privacy risk likelihood for the organisation. However, the lifting to brand level is simple in the sequel of this manuscript (Section 4.4.4) because of the underlying privacy objective: all threats reduce to the threat of personal data breach, and its likelihood is constant over the various assets because these ultimately are types of data.

A similar lifting process should occur for the impact. The overall privacy risk impact could be derived as a generalised sum of the impacts of all threats on all assets. This gets both simpler and more complicated in our specific application to car brands (Section 4.4.3). If, on one hand, only personal data breach applies, on the other hand, the impact on each asset shall be a weighted version of the asset value.

The STRIDE approach

The best-known threat-oriented risk estimation process is STRIDE [87], developed by Microsoft and relying on 6 categories of threats: Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, Elevation of privilege. The name is the simple acronym of the category names.

The focus on threats implies that threat identification may have to be reiterated to recognise all possible threats per category. It is also convenient to highlight which assets would be affected (by non-negligible impact). The general considerations about subjectivity through likelihood assignment, and about impact of a threat on an asset as bound to the asset value, continue to apply here. However, the risk level is assessed for the given threat in the domain, independently from a specific asset because the threat impact is considered holistically on all assets, as the formula shows:

$$\begin{aligned} &Risk(threat) = \\ &\quad Likelihood(threat) \times \\ &\quad \sum_{asset} Impact(threat, asset) \end{aligned} \tag{4.2}$$

Therefore, such threats refer to the application domain in general. Lifting this formula at the level of the organisation may only require a verification of which risks specifically apply to the organisation and then add up their levels. However, this could be complicated by the number of threats, as shall be seen below (Section 4.6.2) over car brands.

4.2.4 Risk evaluation

Risk estimation provides the necessary input to the risk evaluation process. The purpose of risk evaluation is to assist in making decisions about which risks need treatment and their implementation priority.

To evaluate risks, the organisation compares the risk levels with a set of criteria defined during the initial context establishment and possibly reviewed through the various steps of the assessment. The aim of the evaluation is to match the risk levels to the criteria to decide whether the levels meet the criteria. For example, a risk level of 50 meets criteria stating a minimum level of 40. In turn, criteria, which should also comply with legal and regulatory requirements, could derive from an *absolute* or a *relative* approach. An absolute approach yields firm criteria such as "the minimum level is 40". Alternatively, a relative approach sees a prioritisation of the risks by ordering the risk levels, grouping them coherently and assigning them some urgency for some treatment. Relative evaluation approaches shall be applied below.

For example, a relative approach relying on a chromatic scale red-orange-yellow-green may group the risk levels into a colour depending on how many non-zero risk levels are available, as represented in Table 4.1. Clearly, at least 4 risk levels are needed for the 4 groups to become meaningful.

Then, either approach may assign an urgency for a treatment to the various groups of risk levels by leveraging the chromatic scale as follows:

- *Green* or *low risks*: risk may be treated by acceptance;
- *Yellow* or *modest risks*: risk must be treated only if additional cost-benefit analysis is carried out; treatment by acceptance is possible;

Table 4.1: Example application of relative criteria to group risk levels for risk evaluation

Number of				
Risk levels	Green values	Yellow values	Orange values	Red values
1				1
2			1	1
3		1	1	1
4	1	1	1	1
5	1	1	1	2
6	1	1	2	2
7	1	2	2	2
8	2	2	2	2
9	2	2	2	3
10	2	2	3	3
...				

- *Orange* or *tangible risks*: risk must be treated soon; treatment by acceptance is prohibited;
- *Red* or *high risks*: risk must be treated as a matter of urgency; treatment by acceptance is prohibited.

4.2.5 Risk treatment

Risk evaluation inspires risk treatment, more precisely, the applicable decisions on the risk treatment strategies and methods to take.

The treatment involves selecting one or more options to face the evaluated risk levels. Typical options are the application of new controls that reduce the likelihood or the impacts, the transfer of risks to other parties or the acceptance of risks.

Options should be selected considering not only the outcomes of the estimation phase but, if possible, also the expected cost to implement those options and the expected benefits stemming from those options. Risk treatment options are not necessarily mutually exclusive or appropriate in all circumstances. Organisations can benefit from a combination of options, also taking into due account the applicable legal and regulatory requirements.

4.3 Bootstrapping the privacy risk assessment

This Section instantiates the risk assessment over 11 real-world car brands [14], thus taking both approaches discussed above. Precisely, it refers to the yellow, leftmost box of the framework flowchart as represented above (Figure 4.1).

4.3.1 Selecting the car brands

In this study, the target car brands are chosen in terms of market shares. The top ten best-selling car manufacturers during the first quarter of 2019, according to "Car Sales Statistics" [11] are: Volkswagen, Renault, Peugeot, Ford, Opel, Mercedes, BMW, Audi, Skoda, Toyota. In addition, a specific brand widely recognised as a pioneer of electrification, Tesla, has been added.

4.3.2 Sourcing the data categories

Costantino, De Vincenzi and Matteucci studied the policies of the to car brands and pinpointed the data categories that each brand declares to collect [39]. Table 4.2 summarises them. This information is very relevant to bootstrap both flavours of our risk assessment process, as shall be seen below.

Table 4.2: Data categories that each car brand declares to collect [39]

	DC1	DC2	DC3	DC4	DC5	DC6	DC7	DC8	DC9	DC10
Manufacturer	Personal	Geolocation	Driver's Phone	Purchase	Offences and Violations	Driver's Behaviour	Vehicle Status	Surrounding Vehicle Environment	Voice and Messages	App Usage
Volkswagen	✓	✓	✓	✓		✓	✓		✓	✓
Renault	✓	✓		✓			✓		✓	
Peugeot	✓	✓	✓	✓			✓		✓	
Ford	✓	✓	✓	✓		✓	✓		✓	✓
Opel	✓	✓	✓	✓			✓			✓
Mercedes	✓	✓	✓				✓		✓	✓
BMW	✓	✓					✓	✓		✓
Audi	✓	✓	✓	✓			✓		✓	✓
Skoda	✓	✓			✓	✓	✓	✓	✓	
Toyota	✓	✓	✓	✓					✓	
Tesla	✓	✓	✓	✓	✓	✓	✓	✓		✓

4.3.3 Identifying the assets

The various types of data treated by modern cars are the privacy-relevant assets. The current technology landscape in the automotive domain was independently analysed by examining the relevant state of the art and identified the following assets:

- **Personally Identifiable Information**: any data that could potentially be used to identify a particular individual (such as full name, date and place of birth, driver's license number, phone number, mailing and email address).

- **Special categories of personal data**: about the driver, e.g. racial or ethnic origin, political opinions, religious or philosophical beliefs, trade union membership, genetic data, biometric data, data concerning health or data concerning sex life or sexual orientation (GDPR art. 9).
- **Driver's behaviour**: driver's driving style e.g. the way the driver accelerates, speeds up, turns, brakes.
- **User preferences**: data regarding cabin preferences, e.g., seating, music, windows, heating, ventilation and air conditioning (HVAC).
- **Purchase information**: financial information of the users such as credit card numbers and bank accounts.
- **Smartphone data**: data that the vehicle and user's smartphone exchange with each other via the mobile application and short-range wireless connections such as WiFi and Bluetooth (contact book, phone calls, text messages).
- **GPS data**: vehicle geo-location history and route tracking.
- **Vehicle information**: vehicle information such as car maker, model, vehicle identification number (VIN), license plate and registration.
- **Vehicle maintenance data**: data on the maintenance and status of vehicle components such as kilometres travelled, tyre pressure, oil life, brake, suspension and engine status.
- **Vehicle sensor data**: data analysed and calculated by car sensors such as distance sensors, crash sensors, biometric sensors, temperature sensors and internal and external cameras.

To reduce bias, asset identification work was completed prior to learning the categories identified by our colleagues[39]. The work then continued by valuating the assets on a scale from 1 to 5 upon the basis of the sensitiveness of data — the top value in fact was only assigned to the special categories of personal data. The outcome is in Table 4.3.

Table 4.3: List of privacy-related assets and their valuation

ID	Name	Value
A1	Personally Identifiable Information	4
A2	Special categories of personal data	5
A3	Driver's behaviour	2
A4	User preferences	2
A5	Purchase information	3
A6	Smartphone data	4
A7	GPS data	3
A8	Vehicle information	2
A9	Vehicle maintenance data	3
A10	Vehicle sensor data	4

4.4 Taking the ISO approach on car brands

This Section discusses the blue, top-central box of the framework flowchart as represented above (Figure 4.1). The goal of the present exercise is to compare the various car brands with each other in terms of the overall privacy risk affecting a brand. It could be seen above that the ISO approach is based on assets, therefore this effort rests on the relevant assets and their values to calculate impact and likelihood that threats materialise.

4.4.1 Identifying the threats

A number of threats exist for the privacy of (the data of) people. For example, personal data could be illicitly disclosed to anyone that the data owner did not intend as a recipient of the data; similarly, data could be altered or destroyed. Similar scenarios are instances of the overarching applicable threat: a *personal data breach*. This is precisely defined by GDPR art. 4.12 as "a breach of security leading to the accidental or unlawful destruction, loss, alteration, unauthorised disclosure of, or access to, personal data transmitted, stored or otherwise processed". Therefore, also in the interest of brevity, the sequel of this Section refers to a data breach as the only outstanding threat.

4.4.2 Matching assets to data categories

Prior to calculating the impacts of a data breach in the various cases, the identified assets need to be matched with the data categories recalled above. Three matches are obvious, quite a few need some grouping before a matching can be drawn, and no matching is possible in one case.

- Obvious matches. PII, Purchase information and GPS data have an obvious one-to-one correspondence with the data categories as follows:
  - A1 to DC1
  - A5 to DC4
  - A7 to DC2
- Non-obvious matches. These do not have an obvious one-to-one correspondence, e.g. "Smartphone data" includes the categories of data relevant to

the driver's smartphone, i.e. "Driver's phone", "App usage" and "Voice and Messages". Another example is the declared data type "driver's behaviour", which is associated with the assets "Driver's Behaviour & User preferences". The other matches are shown here:

  - A3 to DC6
  - A4 to DC6
  - A6 to DC3, DC9 and DC10
  - A8 to DC7
  - A9 to DC7
  - A10 to DC5 and DC8

- Impossible matches. It turns out that no car brand declares to collect special categories of personal data, hence the asset A2 cannot be matched to any data category.

4.4.3 Calculating the impacts

The next step is to assign a weight to each asset depending on whether it appears in the policy of a car brand, represented as a data category. A weight allows us to scale down the value of an asset when that asset matches more than one data category among those mentioned in the policy. Therefore, the associations between assets and data categories discussed above play a crucial role here. For example, asset A6, smartphone data, relates to the three data categories DC3, DC9 and DC10 (that is, driver's phone, voice and messages and, finally, app usage). Therefore, if a given policy only treats one of those three data categories, then the asset value would be scaled down to a third. The precise assignments of weights are shown in Table 4.4.

Table 4.4: Assigning weights to data categories

Asset ID	Matching data category	Asset value	Asset weight if data category is declared	Asset weight if data category is undeclared
A1	DC1	4	1	
A2	-	5	-	
A3	DC6	2	1	
A4	DC6	2	1	
A5	DC4	3	1	
A6	DC3	4	0.33	
A6	DC9	4	0.33	0
A6	DC10	4	0.33	
A7	DC2	3	1	
A8	DC7	2	1	
A9	DC7	3	1	
A10	DC5	4	0.5	
A10	DC8	4	0.5	

For each car brand, upon the basis of the data categories it claims to collect, we calculate a numerical value representing the impact that a data breach would have on each asset by weighting its value as explained above:

$$Impact(asset) = Value(asset) \times Weight(asset)$$

Clearly, applying this formula to all assets of all brands required deriving the right weights, hence tight consideration of all information given in the tables above, particularly in Table 4.2. Impact at asset level could then be lifted at brand level by adding it up over all assets:

$$Impact(brand) = \sum_{asset} Impact(asset)$$

The values obtain are in a range from from 11.3 to 21.6, meaning that the higher the value, the higher the impact that a data breach would have on the car brand. Notably, the top impact of 21.6 concerns Tesla, which collects all data categories discussed above with the only exception of "Voice and message". The impact on Tesla, represents well the large amount of data about both the vehicle and its driver their cars collect. The full list of impacts is given in Table 4.6.

4.4.4 Assigning the likelihood

As already noted, establishing the likelihood of the manifestation of a threat generally is a subjective process. Bias is normally reduced by focus groups and, most importantly, by bringing existing evidence of prior manifestations of the same threat. For this work, classic web searches were used as a source of relevant information, constructing query pairs such as "brand name, keyword", with "brand name" ranging over the 11 car brands and "keyword" ranging over the set *breach*, *vulnerability*, *exploit* and *attack*. Therefore, a total of 44 web searches were carried out and the results studied. Those considered relevant are shown in Table 4.5.

Through the process of assigning likelihood values for a data breach on each asset, it became apparent that each asset gets the same value because all assets are data that the vehicle collects and treats. Therefore, the process produced one likelihood value per brand, precisely derived by counting the number of relevant hits for the brand and then augmenting it by one. In consequence, brands that produced

Table 4.5: Relevant web search hits

Brand	**Relevant hits**
Volkswagen	Breach [102], Vulnerability [94, 100],
Renault	Breach [55]
Peugeot	*none*
Ford	Breach [113], Vulnerability [94]
Opel	*none*
Mercedes	Breach [78], Vulnerability [5]
BMW	Breach [34, 106], Vulnerability [101], Exploit [118]
Audi	Breach [102], Vulnerability [100], Exploit [1]
Skoda	*none*
Toyota	Breach [33], Attack [61], Vulnerability [117]
Tesla	Breach [29, 69], Attack [80], Vulnerability [128]

no relevant hits get a unit likelihood, hence the impact gets preserved as is in the estimated risk level. The full list of likelihood values is given below, in Table 4.6.

4.4.5 Estimating and evaluating the risks per car brand

Once the likelihood and the impact of a personal data breach are available for each brand, the risk that the breach materialises can be customarily estimated by multiplying likelihood and impact. Table 4.6 shows the complete findings for all car brands. The risk levels are calculated using Formula 4.1 from Section 4.2.3, and span over 10 non-zero values. By taking a relative evaluation approach as discussed in Section 4.2.4, risk levels are evaluated through a chromatic scale of 3 red values, indicating a high risk, 3 orange values, meaning a tangible risk, 2 yellow values, signifying a modest risk, and 2 green values, for a low risk.

Table 4.6: Privacy risk levels per car brand — ISO approach

Manufacturer	**Impact**	**Likelihood**	**Risk**
Tesla	21.6	5	108.0
Volkswagen	19.0	4	76.0
Audi	17.0	4	68.0
BMW	13.3	5	66.5
Ford	19.0	3	57.0
Toyota	12.6	4	50.4
Mercedes	14.0	3	42.0
Renault	11.3	2	22.6
Skoda	17.3	1	17.3
Peugeot	15.6	1	15.6
Opel	15.6	1	15.6

It is apparent that a high risk affects, in order, Tesla, Volkswagen and Audi; a tangible risk affects BMW, Ford and Toyota; a modest risk affects Mercedes, Renault and Skoda; finally, a low risk affects Peugeot and Opel.

4.5 Adapt the STRIDE categories to the automotive sector

This Section summarises the findings of a privacy risk assessment exercise conducted over the automotive domain by taking the STRIDE approach [38].

4.5.1 Spoofing

An example of identity spoofing is to illegally access and then use another user's authentication information, such as username and password.

Similarly to traditional websites, **mobile apps** can also be targets for spoofing attacks. The purpose in creating applications that try to reproduce the appearance of the original application in the best possible way is to deceive the user. An application that is graphically faithful enough can mislead users, who will enter their data. Once the criminals obtain victims' credentials, all they need to do is to install the corresponding app on their own phone and login as the victim. Incriminated applications are usually accompanied by phishing emails or other social engineering techniques to have the application installed on the victim's smartphone [52]. The likelihood of success of this threat increases when stronger authentication measures, such as Two-Factor-Authentication, are not implemented by the genuine app. The assets involved in this threat are mainly personal data of the victim, data accessible from the smartphone on which the malicious app is installed, driver's behaviour, user preferences and also payment data saved in the genuine app account. The likelihood of this threat is high.

As for **smart keys**, a possible spoofing attack is related to brute forcing techniques, that is, to try every possible signal combination to unlock the victim's vehicle. This type of attack, as well as brute forcing in general, does not have much chance of success when the required signals are encrypted or too complex to replicate. The likelihood of success of the attack is therefore low.

Smart keys are also subject to cloning attacks. The signal may be amplified and intercepted for later use [56]. This not only allows a potential attacker to enter the vehicle, but also allows him to start the engine without arousing suspicion [63]. The likelihood of attack is medium because the attacker needs to be nearby the victim's key fob to clone the signal.

Modern infotainment systems have a built-in navigation system that can be the target for **GPS Spoofing** attacks. GPS Spoofing attacks might aim to fake the turn-by-turn navigation to guide the victim to a wrong destination without being noticed [148]. The tools to do this are quite cheap, a few hundred dollars. A potential attacker just needs to operate a radio transmitter powerful enough and send a false but technically correct GPS signal in order to set aside satellites and make all nearby GPS receivers calculate the wrong coordinates. Receivers do not have the technical means to determine the direction of the signal, so they are not aware that the signal comes from a completely different source. The likelihood of this threat is medium because a potential attacker needs to be near the victim's car to perform the attack.

V2X communications are based on wireless connections. VANET (Vehicle Ad-hoc Network) connections use both long-range (3G, LTE) and short-range (Bluetooth and WiFi) connections. Since they are wireless communications, everyone can receive the signals and messages that are sent by the vehicles. The lack of data regarding the freshness of the messages, such as timestamps and nonces, can allow possible replay attacks of previously intercepted messages. These replay attacks can alter the behaviour of the vehicle, for example by replicating a message from a Road Side Unit that communicated the position of the other vehicles or the traffic situation. Also, a vehicle can identify itself as an emergency vehicle and deceive the other cars that will have to facilitate its route. However, since the DSRC and C-V2X protocols provide encryption and authentication by default, it is not easy for an attacker to carry out similar attacks. The likelihood of this threat is low.

4.5.2 Data tampering

Data tampering involves the malicious modification of data. Examples include unauthorised changes made to persistent data, such as that held in a database, and the alteration of data as it flows over a network.

The **infotainment systems** may be subject to malware designed to damage the system by compromising the data it contains. Both input peripherals and applications provide a possible attack surface for injecting malware into the software system. Automatic execution of scripts on USB devices [122], malware hidden inside audio tracks on CDs [120] and even integrated browser navigation [32] can be exploited as transmission media for malicious software. Considering the numerous access points available for spreading malware and the experience of successful exploits, the likelihood of this threat is high. The assets involved in the infotainment system are among the most sensitive in terms of user privacy. Thus, the possibility of integration with smartphones through dedicated applications or through Bluetooth connectivity makes data such as contacts list, call logs, text messages and email addresses accessible within the infotainment system. In addition, data regarding the geolocation of the vehicle and the routes taken are also managed by the infotainment system thanks to the built-in navigation system. Finally, as the “in-car payments” phenomenon advances, to facilitate the purchase of goods and services such as fuel and parking lots, data such as credit card numbers and bank accounts will be increasingly present inside the vehicle [98]. A mismanagement of their processing can lead to data leaks such as driver’s behaviour and user preferences [36, 27].

Malware can also affect **mobile applications** that are vulnerable to code injection. In fact, mobile applications that are not obfuscated, in particular Android applications, can be decompiled and recompiled. Once the application is decompiled, it may be possible to edit and add arbitrary code to the source before recompiling everything into a new APK file. This APK file can then be uploaded to third-party stores to distribute it using social engineering techniques. Unlike spoofing attacks on mobile apps, this threat does not create fake applications but uses the genuine application as the basis for the malicious one. Once the victim has installed the application, the attacker will be able to get all the information processed by the application including the victim's personal information and preferences, her synchronised vehicle information (also driver's behaviour) and (as many applications allow to locate the vehicle) GPS data [58]. The likelihood of this attack is high, considering that many official applications do not implement security measures [82].

The OBD-II standard specifies the possibility to **reprogram ECUs** through the connector. The firmware image can then be retrieved through the update channels, which are mainly two: Over the Air update and offline update through OBD-II port. These images can be analysed to find possible vulnerabilities or change the behaviour of the ECU in certain situations. A malicious reflash of ECU firmware can compromise the integrity of vehicle data (maintenance and sensors). In the worst case scenario, it is possible to "break" an ECU by forcing the installation of incompatible firmware, causing ECU malfunctioning. The likelihood of attack of this threat is not high because the upgrade packages are likely to be encrypted, and the ECU performs an integrity check before proceeding with the upgrade. In addition, a potential attacker needs to know the correct CAN frame to initialise the firmware upgrade process.

The **CAN** bus is very vulnerable to data tampering. Considering the broadcast nature of communications and the lack of integrity checks, it is possible that a malicious node may modify the content in transit in the communication channel by altering the frame bits. Considering the broadcast nature of communications and the lack of integrity checks, it is possible that a malicious node disrupts or causes interference that would prevent a specific message from being received correctly. The assets involved concern maintenance data and sensor data exchanged by ECUs. The likelihood of this threat is low as it is not easy to alter a data frame without anyone noticing.

Another problem for the CAN Bus is **frame injection**. Considering the absence of the authentication field, a malicious node could forge and send frames to trigger actions by other ECUs. With physical access to the vehicle or by consulting online databases, it is possible to map the content of a frame to a certain action of the vehicle (e.g. switch on the vehicle lights, change the values displayed on the speedometer, lock the doors of the vehicle and so on). With this knowledge, once the attacker gets access to the CAN Bus, he can take total or partial control of many relevant features of the vehicle. Compared to other CAN related attacks, for this type of threat an attacker needs some more knowledge about the frames that the target vehicle uses. The likelihood of this threat is high.

V2X communications have threats related to data tampering. In Vehicle-to-Vehicle messages, a malicious intermediate node might modify the message, thus vehicles may receive forged information. Also, a potential attacker could interfere with Vehicle-To-Infrastructure communications such as road signs and therefore cause mayhem with autonomous vehicles. Another possible attack is the sybil attack where one vehicle simulates multiple vehicles by using multiple vehicle IDs. Assets involved are vehicle information, sensor data and GPS data. But, as mentioned above, there are security measures applied by the transmission protocols that make the threat difficult to occur. Therefore, the likelihood is low.

4.5.3 Information disclosure

Information disclosure threats involve the exposure of information to individuals who are not supposed to have access to it, for example, the ability of users to read a file that they were not granted access to, or the ability of an intruder to read data in transit between two computers.

The CAN Bus can be a target for this kind of threats because the frames travel in the clear. This means that any message sent by an ECU is readable from any other node connected to the CAN Bus. All messages that travel on the CAN Bus can be intercepted, and this may cause a privacy infringement. This data can be used to define driver's behaviour and user preferences to uniquely identify a driver. In fact, it has been proven that each person has her own driving style, namely the way she accelerates, brakes, faces curves form data that can identify a driver [57]. This threat is likely once the attacker gets access to the CAN Bus, also considering also that all vehicle information (including maintenance and sensor data) travels on the CAN Bus without any security measures.

Vehicle's data can also be obtained from the **On-Board Diagnostic** component. Through the OBD protocol, it is possible to obtain the diagnostic information of the vehicle and the status of its components such as tyre pressure, brake status, suspension status, oil life, etc. The OBD specification lacks authentication and authorisation mechanisms, as specified above due to the *"right to repair"* imposed by the European Union, which means that all vehicle diagnostic information can be freely obtained by connecting a device capable of interfacing with the OBD connector. The OBD-II port could also be used as a CAN Bus sniffer since it has access to the vehicle's CAN Bus. This means that all frames sent by the ECUs can be captured and analysed later, revealing information about sensors, maintenance and driver's behaviour and user preferences. Computers can also be connected to a vehicle's OBD-II port using USB-to-OBD adapters. The ease of obtaining this information increases the likelihood here.

A possible source of risk associated with the infotainment system is the installed **firmware**. There are several methods to retrieve the firmware image of the infotainment system: debug interfaces, memory dump or download from official websites and specialised forums on the Internet [125]. Once the firmware image is obtained, it can be subject to reverse engineering techniques to discover particularly sensitive information such as cryptographic keys and credentials for access to internal and/or external services (e.g. SSH server, FTP, Cloud Provider). If this information is hard-coded in clear text within the firmware code and if the reverse engineering operation is successful, it may be possible to recover these secrets [129, 90]. Once the cryptographic key used to encrypt the information by the infotainment system is obtained, it may be possible to decrypt locally saved files (PII, GPS data, payments information, driver's behaviour and user preferences). Compared to malware injection, reverse engineering requires higher and more targeted skills, so the likelihood of success of such an attack is lower, thus the likelihood of this threat is medium.

Mobile applications interface with the vehicle via exposed **API endpoints**. These APIs allow one to receive information such as tyre pressure status and vehicle status, but also to locate the vehicle in real time. Access to this data should be authenticated. HTTP requests and responses should also be encrypted. Freely accessible anonymous endpoints or poorly protected APIs (for example by means of default passwords [24]) can reveal not only vehicle status information, but also personal data of the vehicle owner and vehicle location in real time [70, 107]. The likelihood of success of this attack is medium, the automatic controls of online stores

(Google Play, Apple Store) are able to detect unsafe connections and refuse to load the application on the store [35].

Unlike the infotainment system, dumping the firmware from an **ECU** is more difficult since the attacker does not have direct read access to the ECUs. ECUs can implement flash read commands from the CAN Bus, therefore, the attacker would need to know the CAN frame to perform this operation. In general, the most complex ECUs can only be re-flashed through authenticated diagnostic sessions by connecting to OBD-II and using proprietary software of the car maker, which is usually restricted to authorised mechanics only. The only way to find out these commands is by analysing the firmware, thus going back to the starting point, so the likelihood is low.

The security of the **manufacturer's server** is also very important. The organisation may expose poorly protected databases due to incorrectly configured Intranet settings and lack of reliable authorisation methods. An attacker might obtain the database endpoint through traffic analysis and leak all the information in the database such as personal (and financial) customer data, customer's driving behaviour and car data. There are many documented data leak stories where customers are victims of personal data breaches [33, 106, 51]. Therefore, the likelihood of this threat is high.

V2X Communications can also be target of threats related to privacy. A nearby attacker intercepts wireless network traffic to steal information such as driving route, owner's name, vehicle sensor data. In particular, the attacker could sniff V2V messages to and from nearby vehicles as well as V2I messages from the Road Side Unit and then analyse traffic information. Vehicle sensor data could be exploited and could cause a serious privacy leak revealing PII and driver's behaviour.

4.5.4 Denial of Service

Denial of Service (DoS) attacks deny service to valid users, for example by making a web server temporarily unavailable or unusable.

It is easy to cause a denial of service on the **CAN Bus** thanks to the frame priority given by its identifier. In fact, flooding the communication channel with high priority frames, possibly with an ID as low as possible, prevents access and sending by the other nodes, causing a denial of service. Thus, the correct functionality of the communication channel is compromised, preventing the transmission of messages, even critical ones (maintenance and sensor data), between the various control units. It may even concern safety if these operations take place while the vehicle is moving. This threat also has a high likelihood of happening once a potential attacker gains access to the CAN Bus.

A potential attacker might also aim to block the owner's **smart key** signal by preventing the expected operation. There are many devices, even low cost ones, that act on a wide range of frequencies in order to increase the likelihood of successful attack. By blocking the smart key signal remotely, an attacker can prevent the owner of the vehicle from locking the doors of the vehicle, thus facilitating the theft of the vehicle. The likelihood of attack is high, especially when a careless driver does not verify the actual locking of the doors.

A **data loss** on automaker's cloud servers could also cause a denial of service. When there are no data backup mechanisms, it is possible for an organisation to experience loss of information when their data is object of targeted attacks. Unprotected databases and untrained employees can compromise data availability. An attacker using social engineering techniques could send seemingly normal emails to the organisation's employees containing malware, such as ransomwares, irreparably damaging customers data without backups. The likelihood of attack is medium.

V2X Communications can also be subject to jamming attacks and denial of service. The DoS attacks comprise a group of attacks that target the network service availability. In a possible scenario, an attacker with physical access to the vehicle could install a device connected to the OBD or USB connector that, once powered, disturbs all wireless signals and interrupts all non-wired vehicle communications. These attacks may severely impact the performance of applications in the vehicular networks. The primary objective of the attackers lie in disrupting the means of communication as well as disturbing normal services so that they are no longer available to legitimate users. An example of DoS attack is the flooding attack, where the attacker intentionally floods the control channel with a large volume of messages, resulting in network disturbances. Such attacks can increase the latency in the V2X communications and reduce the reliability of the network. Jamming only works in geographically restricted areas, within the range of the attacker wireless device and do not impact V2X communications everywhere. The jamming attack does not require any particular knowledge of the semantics of the exchanged messages. The likelihood of this threat is medium as the attacker needs to be near the targets to cause damage.

4.5.5 Privilege Escalation

In this type of threat, an unprivileged user gains privileged access and thus has sufficient access to compromise or destroy the entire system. Privilege elevation threats include those situations in which an attacker has effectively penetrated all system defences and becomes part of the system itself.

A possible consequence of infotainment reverse engineering is the **modification of the firmware**. If the attacker not only manages to decompile a firmware to read its source code, but somehow manages to modify or add arbitrary code, the severity of the threat and its possible consequences increases. By modifying the firmware, it is possible to perform a privilege escalation, i.e., bypass any security measures, create persistent backdoors, and gain complete control of the infotainment system and the data it contains (PII, smartphone data, user preferences, payments methods, location history and driver's behaviour) [59]. However, the likelihood of success is very low, requiring successful reverse engineering, advanced knowledge of low-level languages such as assembly language, and finally a way to redistribute and install the modified firmware without arousing suspicion. For these reasons, the likelihood of success of this threat is very low.

Another possible attack of privilege escalation concerns **rogue devices** connected to the On-Board Diagnostic port. In fact, many external devices such as dashcams and anti-theft systems connect via OBD port to add functionalities that were not initially designed for the vehicle. However, such devices could potentially add and expose more vulnerabilities as they have access to the CAN Bus through the OBD-II port. Direct access to the CAN Bus allows the device to perform more actions than those designed for the device. An attacker may use exposed OBD device interfaces to send arbitrary commands to the CAN Bus [64]. The injection of CAN frames as described above might endanger the lives of the driver and passengers. The likelihood of this threat is high due to the ease of access to the vehicle's network and trusting drivers.

The risk of **insider threat** should not be underestimated. An insider threat is a malicious activity against an organisation that comes from users with legitimate access to an organisation's network, applications or databases. These users can be current employees, former employees, or temporary workers with access to the organisation's physical or digital assets. Insiders can carry out their plans via abuse of access rights and leak customers data such as PII, financial information, location history and customer's driving behaviour. The attacker may try to take advantage of system or application flaws to gain access to resources they do not have permission to access. There are several means by which an employee can become a compromised insider: phishing, malware infection and credential theft. Lack of authorisation levels and incorrect Intranet configuration can help insider threats to do harm.

4.6 Taking the STRIDE approach on car brands

It could be seen above that the STRIDE approach is based on threats. This Section takes that approach to assess the privacy risks of the 11 car brands. It describes the green, bottom-central box of the framework flowchart as represented above (Figure 4.1).

Section 4.2.3 outlines the approach, in particular promoting the 6 threat categories of Spoofing, Tampering, Repudiation, Information disclosure, Denial of service and Elevation of privilege. The current technological landscape in the automotive domain does not raise significant repudiation threats, hence the sequel of this Section only treats 5 threat categories.

4.6.1 Estimating the risks in the automotive domain

Considering the state of the art in the automotive domain, we tailor STRIDE to identify the relevant threats in each category. This takes considerable effort, and a justification on each threat and its likelihood as explained above in Section 4.5. We then estimate the corresponding risks [38] by systematically applying the Formula 4.2 given in Section 4.2.3. The findings are summarised in Table 4.7, where each threat in each category comes with the relevant likelihood, an indication of the assets (from Section 4.3.3) that are non-negligibly affected and, ultimately, the estimated risk level.

Table 4.7 confirms that the top spoofing risk derives from the companion mobile app and that the top tampering risk from infotainment malware. Both risks concern a rather high architectural level, thereby calling for attention at a software middleware level. The top disclosure risk is CAN eavesdropping and the top denial-of-service risk is CAN Bus flooding. Both concern a rather low architectural level, hence calling for additional scrutiny of bus security and internal network separation. The top escalation risk derives from insider threats, coherently with other application domains.

4.6.2 Estimating and evaluating the risks per car brand

Once the privacy risk levels in the application domain are estimated, the next step is to specify such estimations over the individual representatives in the domain, which in this case are the car brands. Following the general approach explained above, we take that step by deciding which of the domain threats applies to which car brand. For example, we need to verify whether the smart key cloning threat applies to Peugeot, as well as all other possible pairs of threat and car brand. When a threat were found to apply to a car brand, we would then burden the car brand with the risk level for that threat.

To verify such matches, we employ classical web searches as a source of relevant information, as done above, precisely by building query pairs as “brand name, keyword”. While “brand name” continued to range over 11 car brands, this time “keyword” ranged over the 5 threat categories. Therefore, we conduct a total of 55 web searches and study the hits, which only partially overlap with those found before through the ISO approach.

A number of hits are relevant, and we study them to decide what threats applied to what car brands. Opel is found to be vulnerable to attacks of smart key bruteforcing and cloning [62].Other car manufacturers, including Audi, Skoda, Ford and Volkswagen, also suffer similar attacks [108].

Both Ford and Peugeot are affected by a data breach [112, 85]. In addition, Volkswagen has been hit by multiple types of attacks of smart key cloning, data breach and at infotainment level [43, 119]. Audi, BMW and Toyota brands share a similar fate, in fact they receive attacks of smart key cloning and data breach [84, 34, 33]. Renault appears to have received reverse engineering attacks and ransomware [67, 55].

Multiple attacks targeting Tesla vehicles have been documented in recent years [37, 32, 27]. Mercedes, which also turn out rather data hungry (Table 4.2), seems to have been struck by serious attacks — mobile app [147] and data breach [77].

The information outlined here is represented in detail through the ticks in Table 4.8, where all threat (categories) are scaled up to the car brands; the table features a sub-table per threat category. The impact that each threat causes is indicated in brackets. Each tick signifies a threat that is confirmed, through the web search hits, to concern a car brand. The rest of the table can be easily understood. The rightmost columns add up the risk levels per car brand; for example, because

Volkswagen is only concerned by two spoofing threats, of risk level 6 and 12 respectively, the brand's total spoofing risk is 18. The bottom line of each sub-table adds up the number of occurrences of each threat; for example, smart key bruteforcing is a spoofing threat that affects 9 brands.

Each sub-table also evaluates the total risk levels in a relative way and represents them through a chromatic scale. However, the number of non-zero different values varies across the tables:

- 5 for the total spoofing risk levels;
- 6 for the total tampering risk levels;
- 4 for the total information disclosure risk levels;
- 2 for the total denial of service risk levels;
- 3 for the total privilege escalation risk levels.

Therefore, the number of values per colour must be reviewed as discussed in Section 4.2.4.

The findings can be interpreted in many ways. Intra-category considerations highlight the most common risks: smart key issues in spoofing, infotainment malware and CAN injection in tampering, CAN eavesdropping in information disclosure, CAN flooding in denial of service and, finally, rougue OBD-II devices in privilege escalation. Also, the brands that get the top risk levels per threat category are apparent.

Inter-category analyses report spoofing threats are most common, with 29 occurrences, followed by information disclosure ones, with 26 events. Out of 5 threat categories, Mercedes is the brand that gets top risk level most of the times, that is, 4, followed by Ford, Tesla and Toyota, with 2 top places each, and Audi, BMW, Peugeot, Renault and Volkswagen, with 1 red risk level.

Moreover, it can be seen that the most frequent threats that ever materialise, with 9 occurrences reported so far, have got to do with smart keys and the CAN Bus. By contrast, there are 7 threats that have never materialised through events, often involving V2X communications.

4.7 Privacy risk treatment in automobiles

As explained above, a typical risk treatment option is to apply measures that limit risks by reducing impact or likelihood of threats. This Section considers the most common threats per category, based upon Table 4.8, and selects the technical components that are found to be most at risk: smart keys, infotainment systems, CAN Bus and OBD-II. It then outlines technical measures that could be applied to reduce the associated risk levels stemming from either of the approaches taken above. This corresponds with the rightmost, purple box in the framework flowchart as represented above (Figure 4.1)

4.7.1 Smart keys

Smart keys are convenient, allowing one to easily open their car and turn on the engine even from a distance. However, they may pose relevant threats. For example, if the range of action is too wide and the signal is not protected, it is possible to capture and replay the signal at a later time, allowing a thief to steal the vehicle effortlessly.

Signals could be encrypted by one-time passwords so as to thwart replay attacks. A trade-off with usability would be the use of multi-factor authentication, which could thwart scenarios of loss or theft of keys. Smart keys could be hardened by reducing the range of action to a few centimetres, so that a vehicle should not start when the key is not inside it. This could be combined with seat sensors for weight so as to prevent ignition when no driver is inside. The key could be equipped with motion sensors to shut it down after a long lapse of time.

4.7.2 Infotainment systems

There are several countermeasures to reduce the likelihood of threats deriving from the infotainment. Over-the-air updates should be deployed as soon as possible to fix discovered vulnerabilities. In order to prevent malicious code execution from USB drives, the system must check the file system of USB devices and mount only supported file systems. Infotainment firmware should make sure that only necessary USB device classes are enabled, specifically with read-only and no-exec mount options.

Moving on to infotainment applications, the system should allow the installation of software only from official and specific sources. To prevent malware injection, it should deny the installation or the update of software downloaded from unofficial online stores or from users' devices. The system should be able to isolate high risk applications into containers or virtual machines because software software isolation adds an additional security layer.

Update mechanisms should be used in order to deploy security updates and fix discovered vulnerabilities. If the infotainment system provides multi-user support, it should implement access control to separate privileges of different users and should require multi factor authentication at least for administrator login.

4.7.3 CAN Bus

The CAN Bus is one of the main targets during car hacking operations, likely due to its broadcast nature, fragility to Denial of Service attacks, lack of source fields and lack of authentication. Once access to the CAN Bus has been obtained, vehicle control is available as the various components communicate with each other using this communication channel. Most critical attacks concern the injection of artificial messages to trigger unwanted actions or the bus saturation with messages (fuzzing) aimed at predicting the behaviour of the ECUs. Any treatment measure that is conceived should also consider the limited resources that vehicles have with respect to computers, such as low bandwidth, memory, computational power and time constraints, although we can expect that such limitations will fade away in the near future. Also, any measures should be as backwards compatible as possible.

Eavesdropping can be prevented by encrypting messages before transmission over the bus, making it impossible for an attacker to understand the messages sent by the legitimate vehicle components. Intra-vehicular communication must be protected using both message encryption (for confidentiality) and cryptographic hashing (for authentication and integrity). The CIA triad might be achieved by a single software module [17]. Unfortunately, cryptographic key management is not easily applicable on a large scale in the automotive field. Furthermore, considering the large lifetime of an average vehicle, any cryptographic key should be strong enough against brute force attacks. Frames that are not authenticated or come from an undefined source should be dropped by the receiver.

In addition, Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) could help identify and prevent most of the known attacks. Anomalies such as bus load, messages with illegal ID, high number of dropper frames may indicate a potential attack in place. If any anomaly is detected, such systems should quickly warn the driver and the car maker. Another possible measure is network segregation, i.e., separate critical and non-safety-critical ECU connections and use gateways to communicate with each other. This measure, however, requires a modification of the network topology.

4.7.4 OBD-II

The OBD-II port is a powerful entry point to a car. Therefore, it should be secured in such a way that only authorised personnel such as car dealers and mechanics may use this port successfully. Therefore, connecting personnel should be authenticated. Also, diagnostic features should be limited as much as possible to a specific mode of vehicle operation.

Operations that are launched via OBD-II must be secure by default, namely they should provide the most secure configuration by default. Connecting devices should provide only minimum connectivity to connect to the car and execute their diagnostic features. Another countermeasure is to put a firewall (usually referred to as “secure gateway” in the automotive domain) on the OBD-II to prevent malicious command injection to the CAN Bus.

Aftermarket components change the security boundary of the vehicle. Segmentation and isolation from the other components should limit the damage a potential attacker can cause, for example separate CAN communications from the network stack and allow applications to send a request only from a list of pre-defined chosen OBD-II commands.

4.8 Related work

The ISO 26262 international standard [71] is among the best known standards for the automotive domain, hence must be recalled here. It concerns functional safety in the overall domain and, in particular, for electrical and electronic systems employed aboard modern cars [114]. That standard is not directly related to our work, which specifically targets a different property, privacy.

Risk assessment has been framed quite a few times in the context of modern vehicles, in the last decade especially. The work by Wolf and Scheibel [144] can be considered a milestone in its pioneering contextualisation in automotive. However, the framework used is rather basic, and the demonstrating example only concerns attacks to ECU firmware. The need to source information to support the assessment is acknowledged through the definition of a questionnaire to collect information.

Macher et al. [86] instrument risk assessment over safety and security by advancing the so called “Safety-Aware Hazard Analysis and Risk Assessment” (SAHARA) Approach. It combines the “Hazard Analysis and Risk Assessment” (HARA) approach to safety risk assessment with STRIDE in a rather intuitive way but is only demonstrated on a very small example. A sibling contribution by Monteuuis et al. [93] is the “Security Automotive Risk Analysis Method” (SARA) approach, which considers attacker’s features such as knowledge, expertise and equipment as explicit parameters of the assessment. This may be useful when a characterisation of a threat with respect to the specificity of the attacker is requested, whereas this is normally captured implicitly through the threat likelihood. The approach is exemplified on two specific threats independently of a specific brand.

It is worth mentioning here that the Forbes magazine [53] recently published an additional argument for investments in cybersecurity risk assessment. More or less at the same time, Wang et al. [127] advanced an abstract framework for automotive cybersecurity risk assessment. Notably, the claimed advantages of the approach include applicability during the vehicle lifecycle as well as support for quantitative risk metrics. However, the manuscript lacks a convincing running application to demonstrate the benefits and overall strengths of the framework. Applicability to privacy is mentioned but scantly accounted for.

While all these works were inspirational for ours, none of them explicitly target privacy or treat it extensively. By contrast, this work distilled out both assets and threats that are relevant to privacy in the automotive domain using a novel, double approach, which is then demonstrated in practice by offering ways to compare the top car brands. Such a treatment at the specific level of car brand also seems a distinctive feature with respect to the existing literature.

4.9 Discussion

Modern cars expose a variety of digital services and process a variety of personal data, at least of the driver's, hence the motivation for the privacy risk assessment framework is discussed and demonstrated in this chapter. By taking both the asset-oriented ISO approach and the threat-oriented STRIDE approach in parallel, and by specifying the details to instantiate them to the automotive domain first and to specific car brands later, we built a privacy risk assessment framework that can be easily applied by anyone to any automotive-based scope.

Executing the framework by sourcing the relevant information through structured web searches produces the following findings. The asset-oriented outcomes call for attention on Tesla, Volkswagen and Audi for the risk of a data breach, while the threat-oriented outcomes are that the top risks affect Mercedes on 4 threat categories and Ford, Tesla and Toyota on 2 threat categories. Additionally, the most common threat per category becomes apparent. It can be appreciated that the parallel approach offers different, complementary standpoints to the assessment, hence augments the understanding of the privacy risk.

Web searches are leveraged here as a (public) source of relevant information, but other sources may become available, for example towards a classified assessment, and be leveraged to review the assigned values and corresponding findings, yet without a need to change the general framework. While technology in general is notoriously data-driven today, so is the specific technology that modern cars progressively embody. This chapter laid the foundations to risk-assess the privacy objective of such common yet complex cyber-physical systems as the latest cars are, ultimately extending, to recall the GDPR, the “protection of natural persons” to those who drive them.

Table 4.7: Application of STRIDE to automotive domain

Threat	**Likelihood**	**Assets-affected**	**Risk level**
SPOOFING			
Mobile App	3	A1, A3, A4, A5, A6	45
Smart Key Brute-forcing	1	A8, A10	6
Smart Key Cloning	2	A8, A10	12
GPS Spoofing	2	A7, A10	14
V2X Message Replay	1	A8, A10	6
TAMPERING			
Infotainment Malware	3	A1, A3, A4, A5, A6, A7	54
Mobile App Malware	3	A1, A3, A4, A7, A8	39
ECU Reflash	2	A9, A10	14
CAN Frame Injection	3	A9, A10	21
CAN Frame Tampering	1	A9, A10	7
V2X Data Tampering	1	A7, A8, A10	9
INFORMATION DISCLOSURE			
CAN Eavesdropping	4	A3, A4, A8, A9, A10	52
Unauthorised Diagnostic Access	4	A3, A4, A8, A9, A10	52
Infotainment Reverse Engineering	2	A1, A3, A4, A5, A7	28
Insecure API Endpoint	2	A1, A8, A9, A10	24
ECU Firmware Dump	1	A8, A9, A10	9
Server Violation	3	A1, A3, A5, A8	33
V2X Eavesdropping	3	A1, A3, A10	30
DENIAL OF SERVICE			
CAN Bus Flooding	4	A9, A10	28
Smart Key Jamming	3	A8, A10	18
Data Loss	2	A1, A5, A8	18
V2X DoS	2	A7, A10	14
PRIVILEGE ESCALATION			
Infotainment Alteration	1	A1, A3, A4, A5, A6, A7	18
Rogue OBD-II Device	3	A9, A10	21

Table 4.8: Privacy risk levels per car brand — STRIDE approach

SPOOFING	Mobile App (45)	Smart Key Bruteforcing (6)	Smart Key Cloning (12)	GPS Spoofing (14)	V2X Message Replay (14)	**TOTAL RISK**
MERCEDES	✓	✓	✓			63
FORD	✓			✓		59
PEUGEOT	✓					45
SKODA		✓	✓	✓		32
OPEL		✓	✓	✓		32
AUDI		✓	✓	✓		32
BMW		✓	✓	✓		32
TESLA		✓	✓	✓		32
TOYOTA		✓	✓	✓		32
RENAULT		✓	✓	✓		32
VOLKSWAGEN		✓	✓			18
Number of affected brands	3	9	9	8	0	29

TAMPERING	Infotainment Malware (54)	Mobile App Malware (39)	ECU Reflash (14)	CAN Frame Injection (21)	CAN Frame Tampering (7)	V2X Data Tampering (9)	**TOTAL RISK**
MERCEDES	✓	✓					93
AUDI	✓			✓			75
BMW	✓			✓			75
TESLA	✓			✓			75
VOLKSWAGEN	✓			✓			75
TOYOTA	✓			✓			75
SKODA		✓		✓	✓		67
RENAULT	✓						54
FORD		✓					39
PEUGEOT				✓	✓		28
OPEL							0
Number of affected brands	7	3	0	7	2	0	19

INFORMATION DISCLOSURE	CAN Eavesdropping (52)	Unauthorised Diagnostic Access (52)	Infotainment Reverse Engineering (28)	Insecure API Endpoint (24)	ECU Firmware Dump (9)	Server Violation (33)	V2X Eavesdropping (30)	**TOTAL RISK**
MERCEDES	✓			✓	✓	✓		118
FORD	✓		✓			✓		113
TESLA	✓		✓			✓		113
AUDI	✓		✓			✓		113
VOLKSWAGEN	✓		✓			✓		113
TOYOTA	✓		✓			✓		113
BMW	✓		✓			✓		113
PEUGEOT	✓					✓		85
SKODA	✓			✓				76
OPEL								0
RENAULT								0
Number of affected brands	9	0	6	2	1	8	0	26

DENIAL OF SERVICE	CAN Bus Flooding (28)	Smart Key Jamming (18)	Data Loss (18)	V2X DoS (14)	**TOTAL RISK**
FORD	✓		✓		46
RENAULT	✓		✓		46
TOYOTA	✓		✓		46
MERCEDES	✓		✓		46
PEUGEOT	✓				28
AUDI	✓				28
BMW	✓				28
TESLA	✓				28
VOLKSWAGEN	✓				28
SKODA					0
OPEL					0
Number of affected brands	9	0	4	0	13

PRIVILEGE ESCALATION	Infotainment Alteration (18)	Rogue OBD-II Device (21)	Insider Threat (42)	**TOTAL RISK**
MERCEDES			✓	42
TESLA			✓	42
RENAULT	✓	✓		39
FORD		✓		21
PEUGEOT		✓		21
BMW		✓		21
SKODA				0
OPEL				0
AUDI				0
VOLKSWAGEN				0
TOYOTA				0
Number of affected brands	1	4	2	7

Chapter 5

Papyrus-based safety analysis automatization

Nowadays, our society relies more and more on the safety of information systems that have various fields of application such as infrastructure (e.g. electricity generation, transmission and distribution), medicine (e.g. heart-lung machines, defibrillator machines) or aviation (e.g. air traffic control systems). All these engineered systems that can pose catastrophic risks to operators and the environment are called safety-critical systems [79]. These systems are constantly growing, so it is important that the development of these systems follows a rigorous system engineering process to ensure that the safety risks of the system are mitigated to an acceptable level. Today's standards define well-established system safety analysis techniques that are widely used during the design of safety-critical systems. Among the most relevant analysis techniques, Fault tree [133] and Failure Modes and Effects Analysis (FMEA) [132] are widely used in the automotive software environment to analyse the propagation of faults in the architectural design. Safety analyses are very often carried out manually and require a huge technical experience, so becoming time consuming and error prone, especially in large complex systems. For this reason, the use of models and tools for automatic analysis of safety-critical systems has attracted increasing interest [149].

This chapter proposes a tool to automate the safety analysis of embedded software systems. With the tool, the software architect (Actor/User) must initially produce a software architectural design specification using the Papyrus SysML model reproducing both static and dynamic design aspects. In particular, the user must represent the functional chain of operations and interactions among architectural elements (e.g. software unit, data structures, functions).

The Chapter is structured as follows: The remainder of the chapter is organised as follows. Section (5.5) describing the related work. Section (5.1) giving details about the methodology applied to the work of this chapter. Then the chapter explains the process followed to obtain the tool outcome, starting from the SysML design realisation phase and continuing to the model processing phase, respectively in Section (5.2) and Section (5.3). The core of the document is defined by Section (5.4) which outlines the entire safety analysis part. Finally, the chapter concludes with some broader evaluations of the results (5.6).

5.1 Methodology

Today, there is a rapid growth of infrastructures which consequently require particular attention from those risks related to functional safety. For this reason, this chapter proposes a novel tool, based on Papyrus [54], an open source tool that allows the creation of models using UML/SysML [143, 116] standards, which aims to automate the safety analysis of systems. This tool provides several user benefits such as:

- performing automatic analysis of the propagation of faults, reduce the manual effort dedicated to safety analysis;
- reducing the probability of human error due to the complexity of the system;
- develop an integrated methodology through open-source solutions where architectural design and fault propagation analysis can be managed on the same tool so reducing the effort of change management.

Data flow or similar diagrams are often used to describe the functional aspects of the software architectural design (e.g. functional or processing chains). The intent is to exploit the functional dependencies included in these diagrams to provide an automatic evaluation of fault propagation in the system. In particular, in this proposal the data flow is modelled as the combination of a write and read operation on the same data structure element made by two different software modules. Several data flow situations may be implemented in a complex software package but this proposal focuses the analysis on:

- write and read operation done by different modules on the same data structure element;
- read as first operation of a module after a write operation done by another sub-functionalities on the same data structure element;
- write operation as output of the nominal function outside the system boundaries.

Each of the above-mentioned situations can be addressed by our methodology by extracting the information included in the architectural specifications diagrams, both in static and dynamic views. The aim of this task is to model each data flow among safety related software modules in order to allow the fault propagation analysis by applying a fault model onto specific functional dependencies. Moreover, due to the complexity of the functionalities, very often it is difficult to identify all the potential source of interferences between multiple functionalities running in the same environment. For this reason, the approach proposed in this chapter can also be used to select those data elements that represent potential root causes of dependent failures because they are used (implicitly or explicitly) by more than one safety functionalities with different integrity.

As prerequisite, the tool requires the SysML diagrams of the system to extract the parameters needed for the subsequent analysis. Clearly, the more accurate the model, the more detailed the analysis will be. Once the system design phase has been carried out, the tool continues with the model elaboration phase which has the purpose of identifying events, i.e. the sequence of relevant operations within a system from the point of view of safety. Subsequently, the tool performs an automated fault propagation analysis. Finally, the tool provides a report with the information of the performed analysis by identifying the events whose failure directly lead to the violation of top level safety requirements and also by selecting potential common cause failures. To evaluate the correct functioning of the tool we decided to run it on a simplified model of a realistic embedded software system.

Figure 5.1 shows the actors and activities performed when using the tool. Specifically, the initial phase is triggered by the user through the "Realise SysML design" action. In addition, the user must define the risk classification (ASIL[1] or QM[2]) to

[1] The Automotive Safety Integrity Level (ASIL) is a risk classification scheme that defines the integrity requirements of a product.

[2] Quality management (QM) indicates that the risk associated with a hazardous event is not unreasonable and therefore does not require safety measures according to ISO 26262.

be associated with software modules and data structures according to the requirements allocated to each unit [71]. Then, the "Model Elaboration" action is started and once completed returns the process completion message to the user. At this point, the user can start the fault propagation analysis and afterwards select the event where the fault shall be injected. Finally, the tool starts the operation that generates the report and returns it to the user.

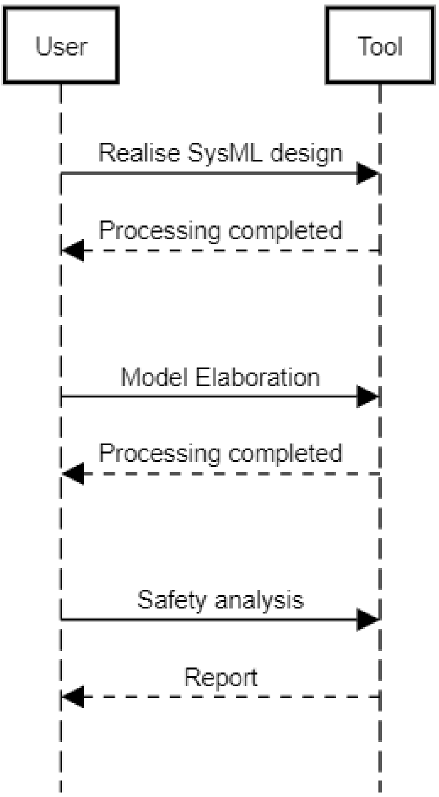


Figure 5.1: Sequence diagram of actors and activities while using the tool

5.2 Process description - Realise SysML Design

The objective of this phase is to construct the SysML diagrams of the system under consideration. Indeed, real-time and embedded systems require detailed system design. Therefore, to avoid systematic faults during design, the architectural specification shall be supported by semi-formal notation as UML/SysML. Due to the diversity of disciplines, it is necessary to customise the modelling to fit the system design. To comply with safety standards recommendations, the user must detail the static and dynamic aspects of the system. To do this, the user defines all components of the system using the “Block Definition Diagram” (BDD) by exploiting the “Internal Block Diagram” (IBD). The IBD plays a very important role in system development as it defines the relationships and interfaces between Units that exchange data either through arguments passed by function calls or implicitly through shared data structures. Therefore, the IBD allows to add a level of detail to the design of the system and to understand the hierarchy among the elements that compose it. An example of a definition of BDD and IBD is shown in Figure 5.2 for a unit called Kernel and a data structure called RQ.

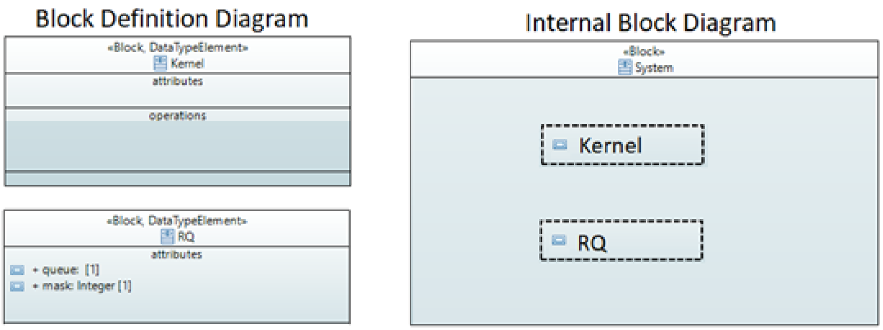


Figure 5.2: Example of Block Definition Diagram and Internal Block Diagram

Once the blocks have been defined, the “Sequence Diagram” can be constructed, to define the execution order of the operations within the system. A fundamental step in constructing the “Sequence Diagram” is to use the blocks defined in the IBD as Lifelines for the Sequence Diagram and to specify which operation is implemented by the module on the identified data structures’ elements. Each software module can either implement a read operation on one (or a group) element, or can modify it with a write operation. Moreover, the proposed tool provides the possibility to tag an operation as “safety mechanism” if the module applies an action to detect potential faults in the data structure element used in the operation.

In case the user needs to describe complex algorithm with more than one independent path inside the same function, combined fragments can be used in the sequence diagram to allow the definition of if/then/else conditions and loops.

After having described static and dynamic aspects of each nominal functionality, the user can draw the interactions between different functions at higher level using the “state machine diagram”. This diagram will be relevant in the following stages to extract the dependencies of lower level functionalities concurring to the same highest level nominal function.

Finally, an example of a Sequence Diagram with the presence of Combined Fragments is shown in Figure 5.3.

The following figure shows an example of a “State Machine Diagram”. From the Figure 5.4 we can notice that the first feature executed is the one called “DeQ” and then the feature “EnQ” is executed.

5.3 Process description - Model Elaboration

The objective of this phase is to process the model built in the design phase and obtain all the information necessary for the last phase, namely safety analysis.

The developed tool, after taking the Papyrus model as input, is able to obtain all the information from the user-defined blocks. An example of the output provided by the tool is shown below.

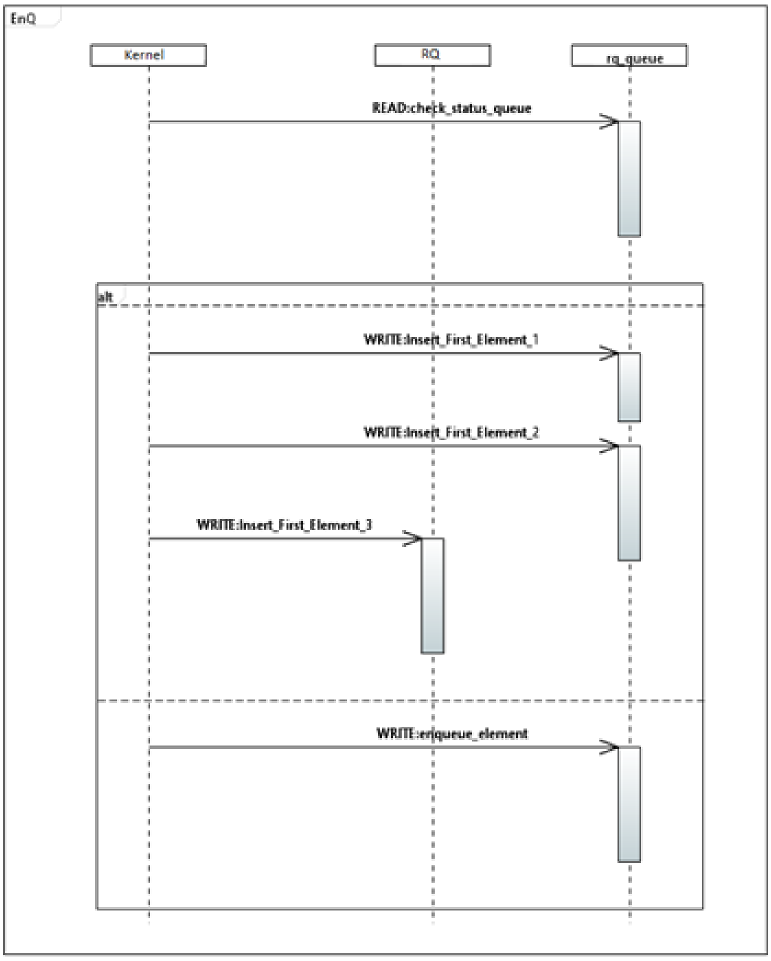

EnQ
Kernel
RQ
rq_queue
READ:check_status_queue
alt
WRITE:Insert_First_Element_1
WRITE:Insert_First_Element_2
WRITE:Insert_First_Element_3
WRITE:enqueue_element

Figure 5.3: Complete example of Sequence Diagram

```
--Starting IBD analysis--

Block Name: RQ

Block Property: queue

Block Property: mask

Block: RQ -

 SubBlock: rq_queue has property: p_head

Block: RQ -

 SubBlock: rq_queue has property: p_tail
```

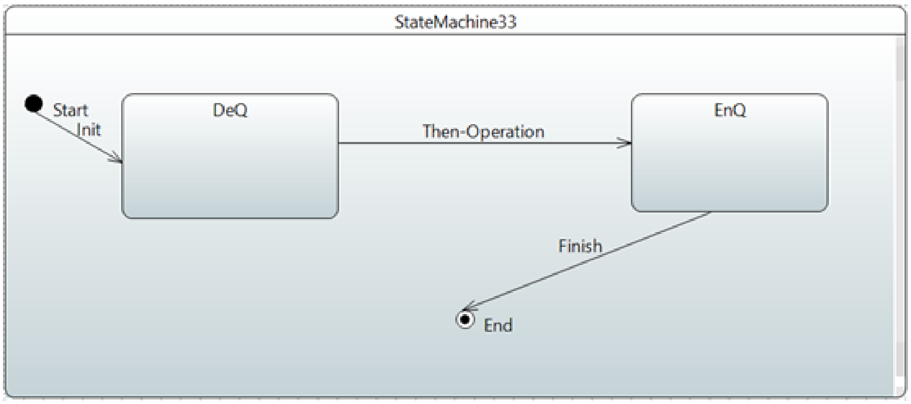


Figure 5.4: Example of State Machine Diagram

To collect all the necessary information for safety analysis and the next fault injection, the tool elaborates the diagrams described in the previous section and extracts the needed characteristics. The output of this elaboration is a matrix as shown in Table 5.1. The matrix consists of 9 columns where:

1. **Struct**, indicates the target data structure where the operation is applied on.
2. **Element**, indicates the element of a data structure on which the operation in column 3 is performed.
3. **Operation**, indicates the operation performed on the Element. This operation may be a Read (R) or a Write (W) operation.
4. **Source Module**, indicates the software module that initiated that operation as defined in the IBD model.
5. **Combined Fragment**, an interaction fragment which defines a combination (expression) of interaction fragments.
6. **Receiver Module**, indicates the name of the receiver module on which an operation is performed.

7. **Event**, sequence of operations that may be subject to fault, i.e. one or more of those explained in Section 5.1. The parameter indicates the ID of the events related to the operation in the same line.

8. **Value EV**, indicates the event status (true as default status, it can be modified to fault to simulate the fault occurrence).

9. **Feature**, indicates whether a line belongs to a specific Sequence Diagram.

Table 5.1: Output matrix generated by model elaboration

Struct	**Element**	**Operation**	**Source Module**	**Combined Fragment**	**Receiver Module**	**Event (EV)**	**Value EV**	**Feature**
rq_queue	p_head	R	Kernel	Null	Null	Null	Null	EnQ
rq_queue	p_head	W	Kernel	IF head==null	Example	0	True	EnQ
rq_queue	p_tail	W	Kernel	IF head==null	Null	1	True	EnQ
RQ	mask	W	Kernel	IF head==null	Null	5	True	EnQ
rq_queue	p_tail	W	Kernel	ELSE head!=null	Null	2	True	EnQ
rq_queue	p_head	R	Example	Null	Null	Null	Null	EnQ
rq_queue	p_head	R	Kernel	Null	Null	3	True	DeQ
rq_queue	p_head	W	Kernel	IF head(next)==null	Null	4	True	DeQ

5.4 Safety Analysis

This Section explains the core of the tool, where the fault propagation is analysed both on its local effect in the faulty C function and also at higher level to the intended nominal functionality.

5.4.1 Creation of Events and Faults relationship

Suppose that a top level safety requirement allocated to the system has been deployed into several software safety requirements and allocated to the architectural elements. The objective of the implemented safety requirements is to detect potential errors and react in time to mitigate their effects in order to avoid the violation of the top level safety requirements.

To analyse the cause-effect relationship of such faults, the tool carries out an analysis of safety-relevant events (e.g. a write to a data structure followed by a read operation), focusing on the data flow between software architectural elements and their interaction. Taking the matrix resulting from Section 5.3 as reference, the tool automatically extracts the Boolean formulas to represent the implementation of a functionality as a combination of events (as described in the previous section). Let's suppose F1 as a safety related feature at module level whose nominal execution is guaranteed if and only if all the underlying events are correctly executed. This situation can be described as follows:

$$F1 = Ev_1 \ AND \ Ev_N \tag{5.1}$$

Starting from the previous formula, the violation of the safety requirements (at module level) can be seen as the negation of the F1 function and correspondingly of the below events. According to Boolean rules, the formula becomes:

$$!F1 =!Ev_1 \ OR \ !Ev_N \tag{5.2}$$

Figure 5.5 shows an example of the application of the equation 5.1. Figure 5.6 shows an example of the application of the equation 5.2.

After having analysed each specific software feature at module level, the model elaboration can be extended to the global nominal function. The relationship among

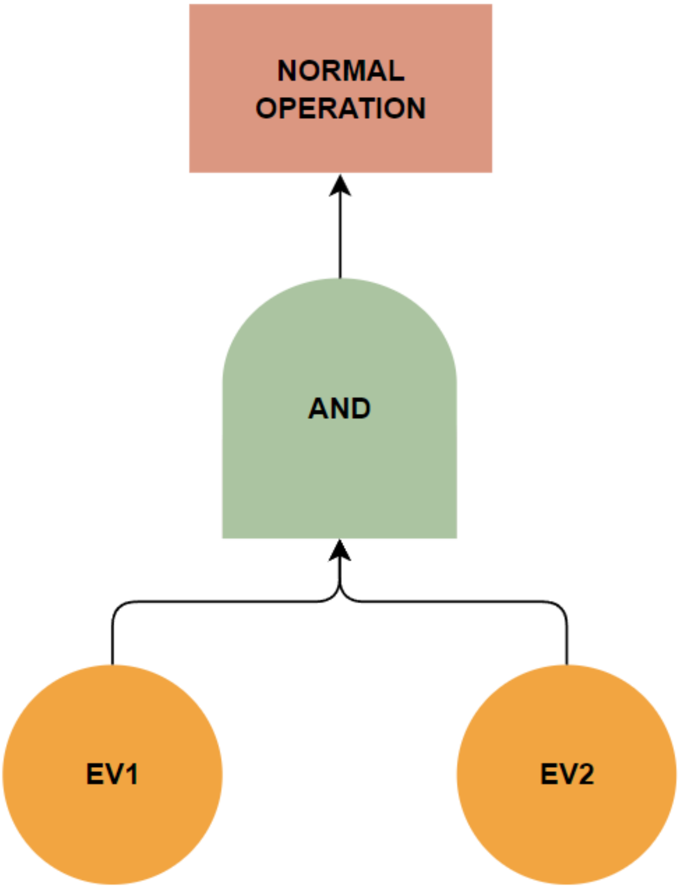


Figure 5.5: Application of the equation 5.1 for specific feature

the events contributing to the nominal function can be expressed by the formula

$$F(global) = F_1\ AND\ F_2\ AND\ ...\ F_N \tag{5.3}$$

where $F_1, F_2, ..., F_N$ represent the occurrence of each F feature (in a nutshell, all the software features shall be executed to complete the global nominal function).

By expanding each Software feature, the tool is able to carry out a global analysis of dependencies between events, taking also into account any safety mechanisms

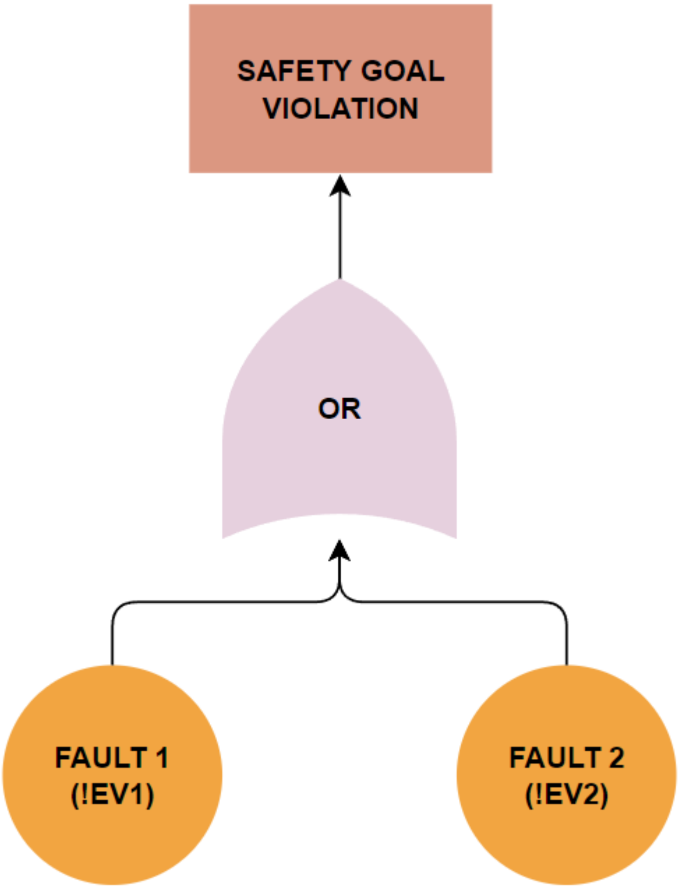


Figure 5.6: Application of the equation 5.2 for specific feature

applied to the model. Let's provide an example by assuming that:

$$F_1 = EV_1 \ AND \ EV_2 \ ; \quad F2 = EV_3 \ AND \ EV_4 \ ;$$

$$FN = EV_5 \ AND \ EV_6.$$

And assuming that EV_2 and EV_5 operate on the same element of the data structure, we can express $EV_5 = F(EV_2)$. So the formula of the global nominal function becomes:

$$\begin{aligned} &F(global) = (EV_1\ AND\ EV_2)\ AND \\ &(EV_3\ AND\ E_4) AND (F(EV_2) AND\ EV_6) \end{aligned} \tag{5.4}$$

By exploiting the same method described above, the negation of $F(global)$ will provide an indication about the dependency of each event's failure with the violation of the top level safety requirements.

5.4.2 Fault injection and effect analysis

The objective of this phase is to introduce the concept of "safety mechanism" into the analysed formulas and carry out fault injections and verify the effect of these operations on the system under test. Finally, the tool provides also the possibility to inspect dependencies among software functionalities by checking Events that are shared among multiple features.

First of all, let's analyse how the insertion of safety mechanisms modify the formulas described in the previous section. By definition, safety mechanism is a technical solution to detect, mitigate or tolerate faults, or control and avoid failures in order to maintain the intended functionality or achieve the safe state. This definition has been simplified into an OR gate between the events and the corresponding safety mechanism implemented to cover the fault. Thus, if we assume that each fault of EV from 1 to N is covered by a safety mechanism, the F1 formula becomes:

$$F1 = (Ev_1\ OR\ SM_1)\ AND\ (Ev_N\ OR\ SM_N) \tag{5.5}$$

$$!F1 = (!(Ev_1)\ AND\ SM_1)\ OR\ (!(Ev_N)\ AND\ SM_N) \tag{5.6}$$

Using the tool, the user can select which event to inject the fault on, thus changing the event status from "True" to "False". Once selected, the tool searches for the event number in the matrix and sets the value of the "Value EV" column to "False". The fault propagation analysis is then performed starting from that event number chosen by the user, then the tool selects all the events linked to that event number and all those Boolean functions impacted by the change of state of the event. Figure 5.7 shows an example of an information flow concerning the application of the formulas, also taking into account a possible safety mechanism.

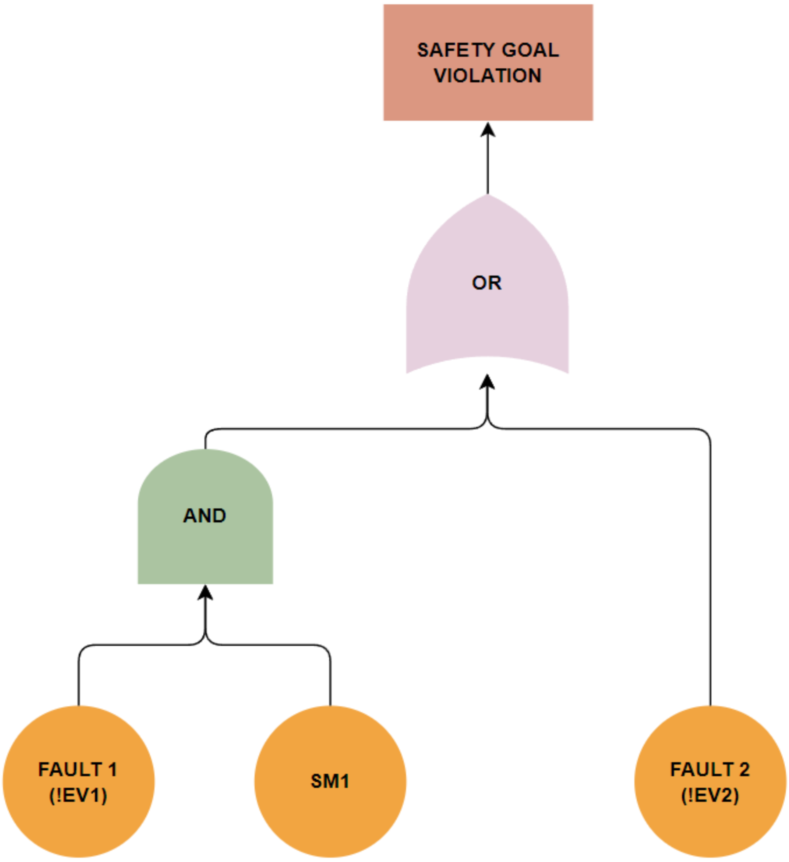


Figure 5.7: Information flow with application of formulas

For what concerns the analysis of dependent failures, the tool extracts which are the dependent events starting from the dependency information extracted in the

"Creation of Events and Faults relationship" phase. As an example, the effect of the interference can be analysed by recalling the formula of the global nominal function defined above,

$$\begin{aligned} &F(global) = (EV_1\ AND\ EV_2)\ AND \\ &(EV_3\ AND\ EV_4) AND\ (F(EV_2) AND\ EV_6) \end{aligned} \tag{5.7}$$

By applying a fault on the dependent events, i.e. those events that can cause cascading failures, the tool will show which are the features impacted by the simulated fault and which are the effect at system level to the nominal functionality. For instance, by analysing the formula it can be seen that EV_2 is the root cause for a cascading failure: in fact, if a fault occurs on $EV2$, it implies the violation of all functions that depend on EV_2 (thus EV_2 and $F(EV_2)$) and so the violation of the safety requirement allocated to the nominal function.

5.5 Related work

In the last twenty years, the safety analysis has been framed several times in the context of systems. One of the oldest works, done by Toola [121], concerns automation safety and the use of safety analysis methods in automation design. However, safety analyses provide information on systematic failures caused by design weaknesses, conditions of faults inducing to the violation of the safety requirements, and therefore the analyses can be used as a starting point when designing systems. Toola, in his paper, points out that a disadvantage of some methods is that multiple failures are not systematically studied.

S. Schreiber et al. [111] propose a knowledge-based approach to support "HAZard and OPerability"(HAZOP) analysis and reduce the manual effort required. The main ideas are to incorporate knowledge about typical problems of automation systems, in combination with their causes and effects, into a rule base, and apply this rule base by means of a rules engine on the description of the automated system under consideration.

Möhrle et al. [92] present an algorithm that generates mappings between failure modes by means of a string distance metric to accelerate the construction of safety artefacts in the early design stages. However, the error susceptibility of the metric and the consequent need for manual revisions make this approach unsuitable for automated safety analysis.

All of these works have been a source of inspiration for ours, but none of them aims at developing a tool to automate the security analysis process from SysML for critical systems.

5.6 Discussion

Nowadays, carrying out safety analyses requires a great deal of effort both in terms of time and human resources. For this reason, more and more solutions are sought to facilitate these analyses by using automated tools that provide accurate results. This article proposes a tool that aims to automate the safety analysis process starting from the SysML diagrams, built through the open source software Papyrus, which allows us to model the data structures, methods and data flow of the system in question. To do this, the tool requires a detailed design of the system under consideration as input. Subsequently, the tool processes the input model and performs the creation of events and faults relationship between the elements of the system necessary for the automated safety analysis which will finally provide a descriptive report. Hence, the report provides the ability to evaluate potential sources of interference between multiple software features to enable optimisation of dependent fault analysis. Clearly, this is a solution that requires a certain effort in the design part of the system under consideration. For this reason, it can be considered as future work, the possibility of extracting the information of the system automatically, for example from the source code of the latter, in such a way as to re-use this solution for those software products that have not been developed following any safety standard.

Chapter 6

Conclusion

The main contributions of this thesis concern the study of cybersecurity, privacy and safety applied to the automotive sector. Our investigation was guided by the observation that modern cars are progressing rapidly in terms of on-board technology. Specifically, all this technology inevitably implies the growth of problems: of cybersecurity since local or remote attackers could compromise cars with cyber attacks, of driver privacy since data in transit inside the car could be exfiltrated and used for profiling purposes.

Chapter 2 studied the implementation of an AUTOSAR-based Basic Software Module, called CINNAMON, which aims to guarantee the main security properties of confidentiality, authentication and integrity while also ensuring the freshness of exchanged messages. For the development of CINNAMON, a real-world scenario was considered, i.e. it was implemented by means of an inexpensive testbed consisting mainly of STM32F407 Discovery boards simulating car control units. In addition, benchmarking of CINNAMON was carried out, which provided very promising results; in fact, CINNAMON adds less than 6μs to generate or validate a protected frame in any of its security profiles.

Chapter 3 studied drivers' concerns about privacy and perceptions of trust. More in detail through crowdsourcing, a representative sample of participants were collected whose responses were then statistically correlated, yielding a wealth of useful information. In fact, through this study it was noted that privacy is poorly understood by car drivers and there is also little trust in security. For this reason, more information may need to be provided to raise awareness, through driver sensitisation, and thus form correct privacy concerns and corresponding appropriate trust perceptions.

Chapter 4 contributed to the development of risk assessment on top car brands through the two best known approaches. One is asset-oriented, as pioneered by ISO 27005 [73], while the other risk assessment approach is threat-oriented and is called STRIDE [88]. The contribution of this chapter is therefore a two-pronged privacy risk assessment of the most popular car brands, based on sales data, which contributes to the development of a privacy-oriented framework for the different car brands. This allowed us to look at the risk from different angles in order to obtain a finer risk assessment analysis.

Chapter 5 contributed to the development of a tool for automating the safety analysis of embedded software systems typical of automotive systems. Since carrying out safety analyses requires a great deal of effort in terms of both time and human resources, the tool developed in Chapter 5 aims precisely at reducing this effort. By means of a detailed system design, the tool processes the input model and performs the creation of the event and fault relationship between the system elements required for the automated safety analysis, which will ultimately provide a descriptive report.

6.1 Future Directions

In this thesis, we have investigated various aspects relating to the automotive field. Most of these topics may have future work ahead of them. Starting with CINNAMON, as future work one could think of its integration on a real car and study its behaviour in order to turn it into a Technology readiness level (TRL) 5 product [142].

Concerning studies on drivers' concerns regarding privacy and perception of trust, a future work could be to carry out data mining and verify all possible inferences provided by the driver data and publish this information to raise awareness among all drivers.

Furthermore, a future development of the work on risk assessment could be to automate this process and apply it to all car brands in order to provide risk assessment data to car companies to mitigate and address their highest sources of risk.

Finally, with regard to the safety analysis tool, a future work could be to enhance the tool by providing it with the ability to extract system information automatically, e.g. from the source code, so as to reuse this solution for those software products that have been developed without any safety standards.

In conclusion, our main intuition is that the automotive environment is complex and its in-depth understanding can enable us to build more sophisticated safeguard mechanisms. In this sense, it is necessary to aim at improving cars on several points of view, so strategies can be: improving cybersecurity applied to cars by reducing attack surfaces, making drivers aware of the amount of data flowing inside a car, showing users the possible profiling activities through data and inference on it, mitigating the risk both from a cybersecurity and a safety point of view.

Appendix A

Graphical representation of results on the study of drivers' privacy and trusts

The following histograms represent the results from the core questions, which were presented in a table format above (Section 3.3).

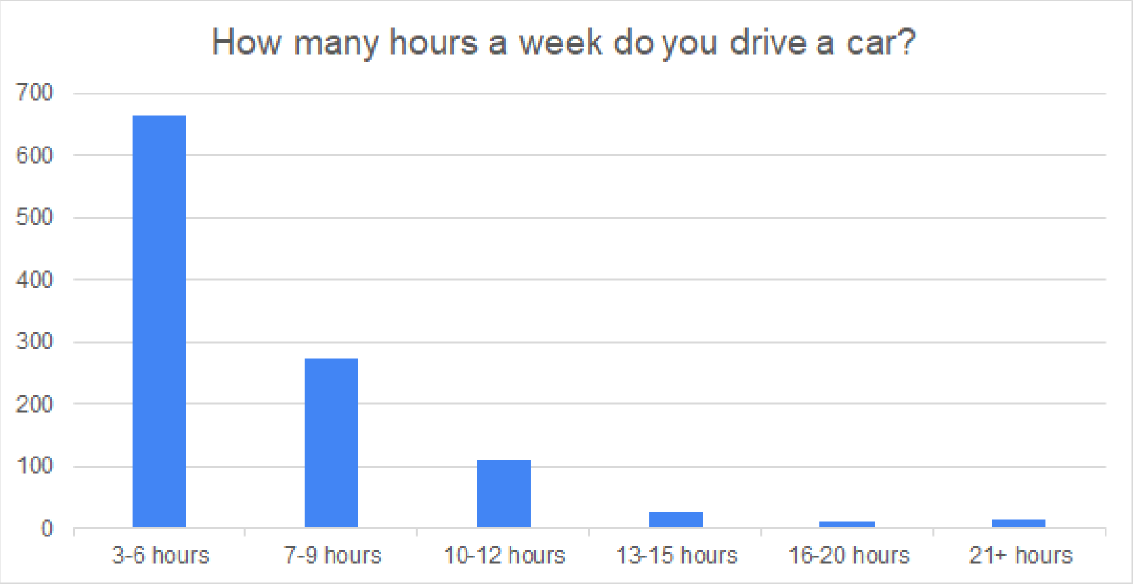


Figure A.1: Time spent driving (N = 1101)

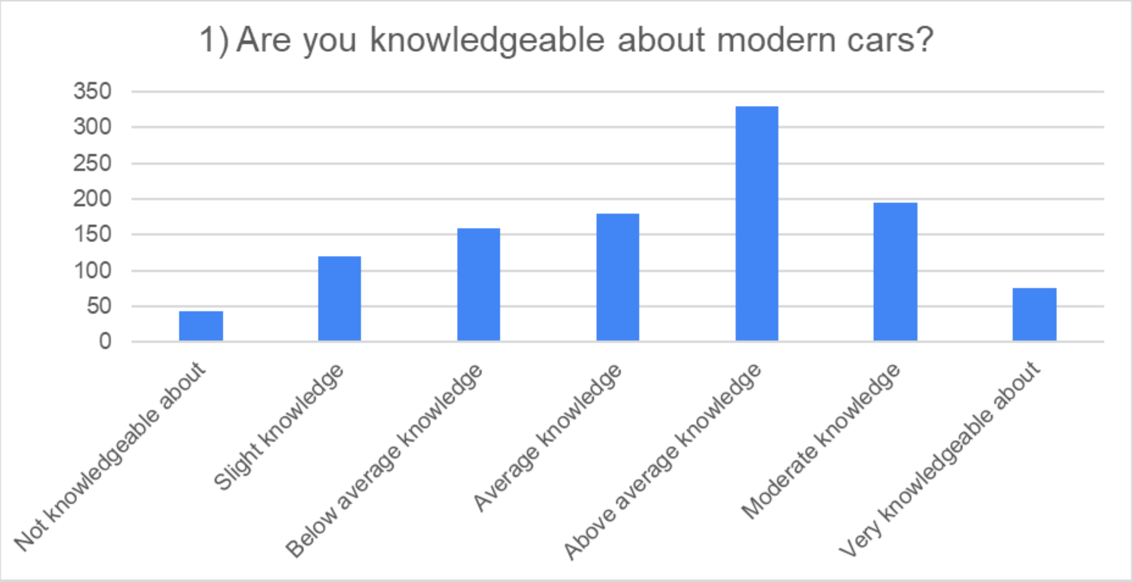


Figure A.2: Results question 1 (N = 1101)

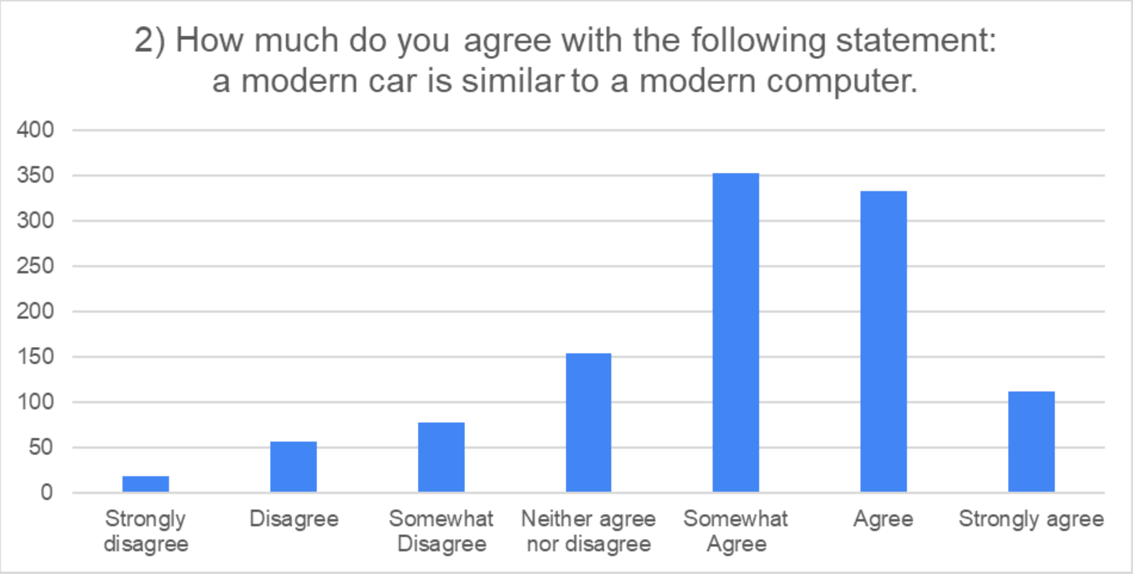


Figure A.3: Results question 2 (N = 1101)

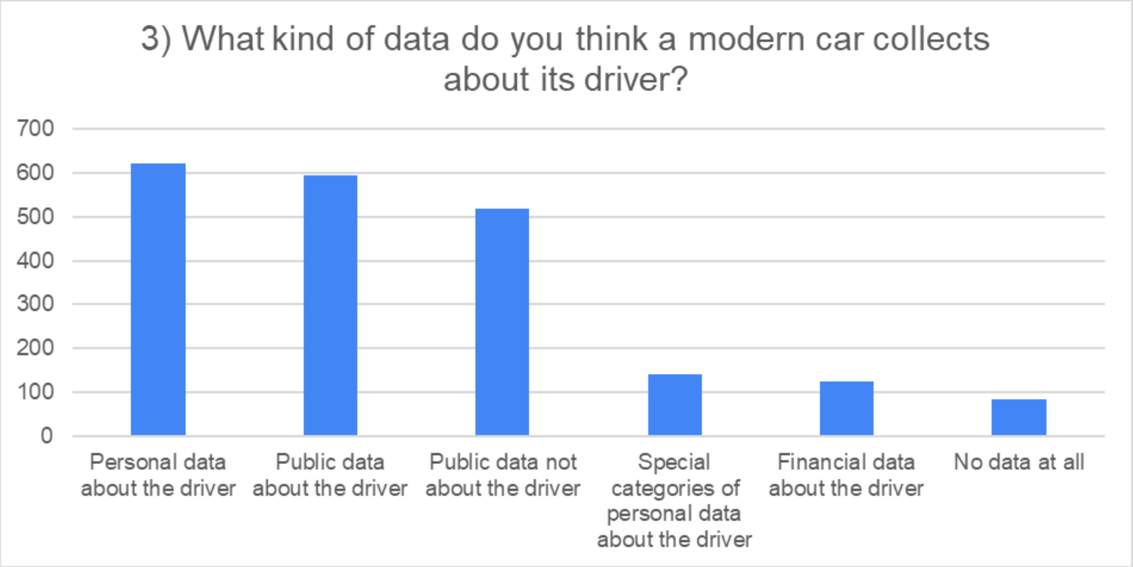


Figure A.4: Results question 3 (N = 1101)

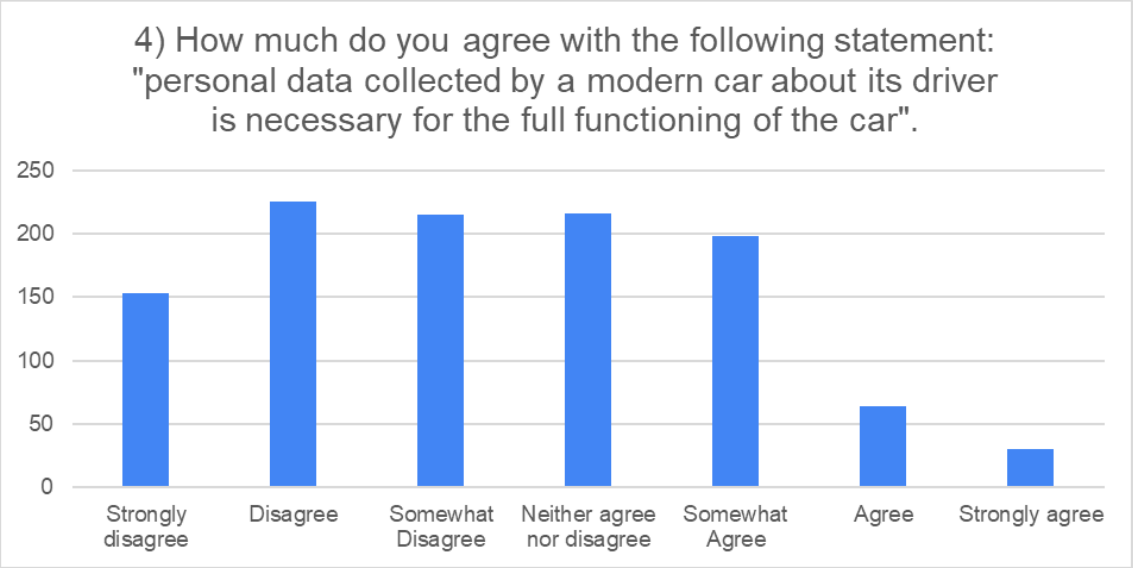


Figure A.5: Results question 4 (N = 1101)

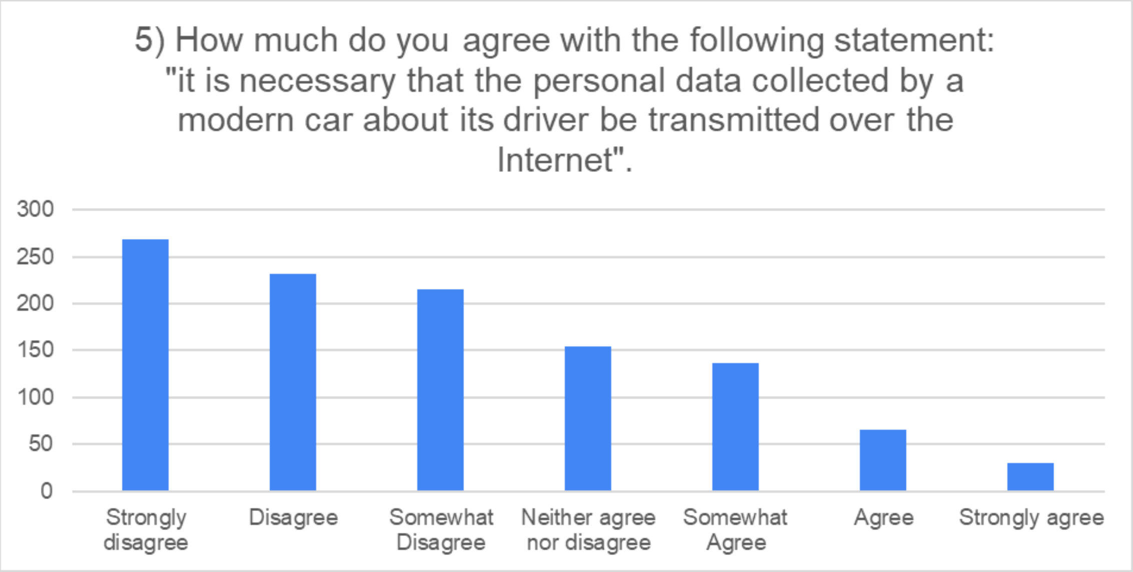


Figure A.6: Results question 5 (N = 1101)

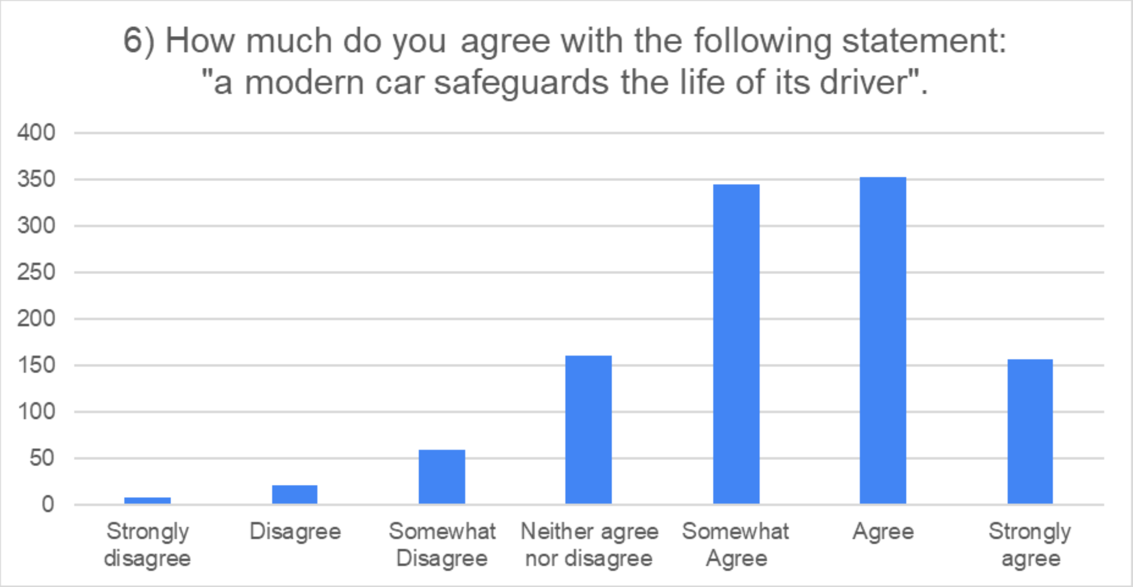


Figure A.7: Results question 6 (N = 1101)

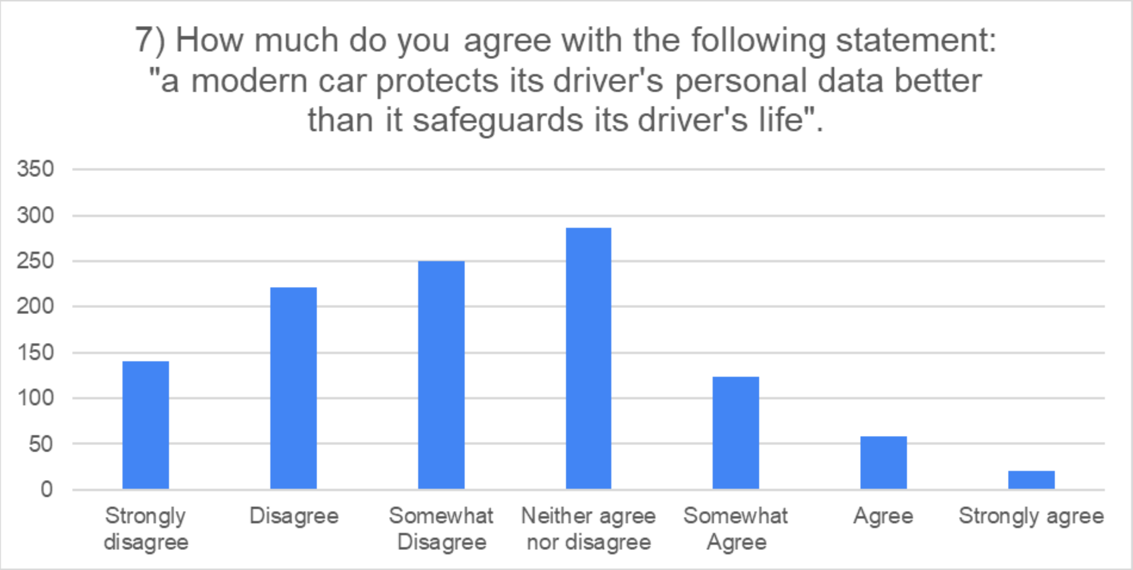


Figure A.8: Results question 7 (N = 1101)

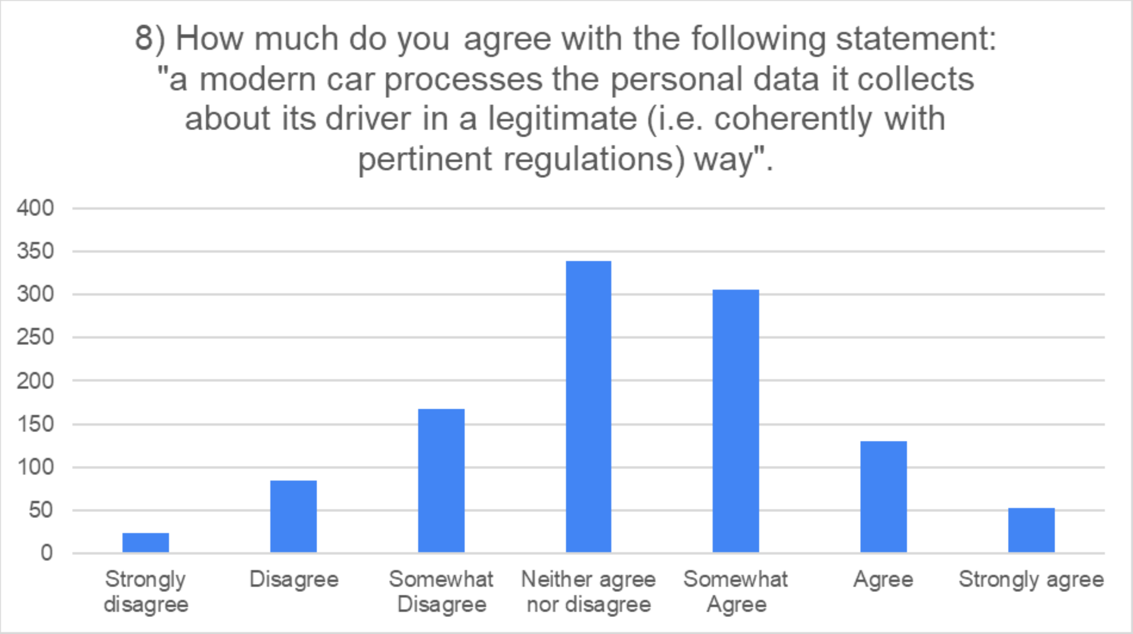


Figure A.9: Results question 8 (N = 1101)

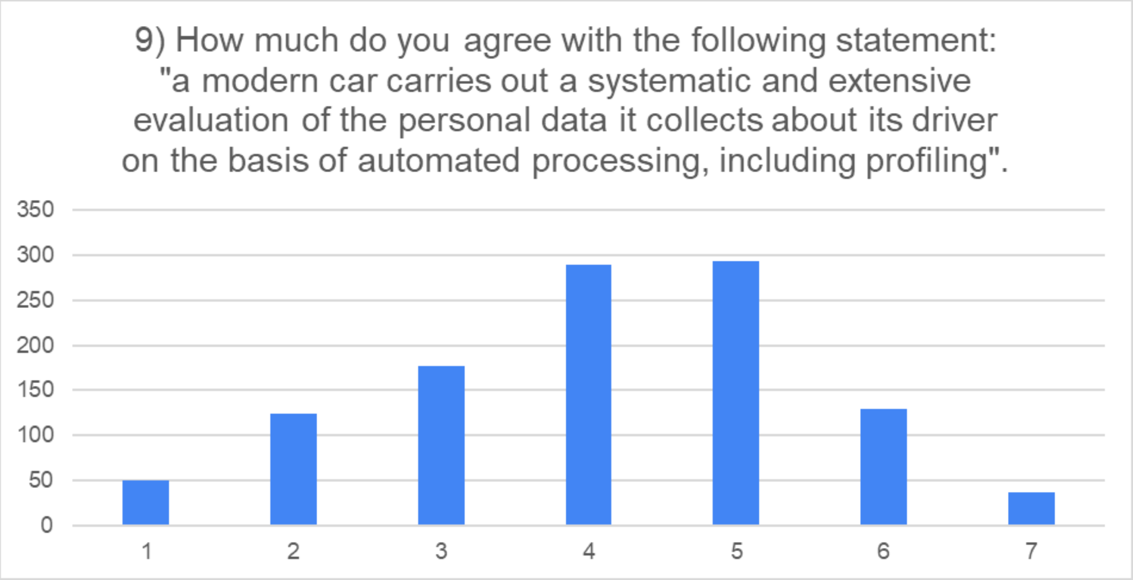


Figure A.10: Results question 9 (N = 1101)

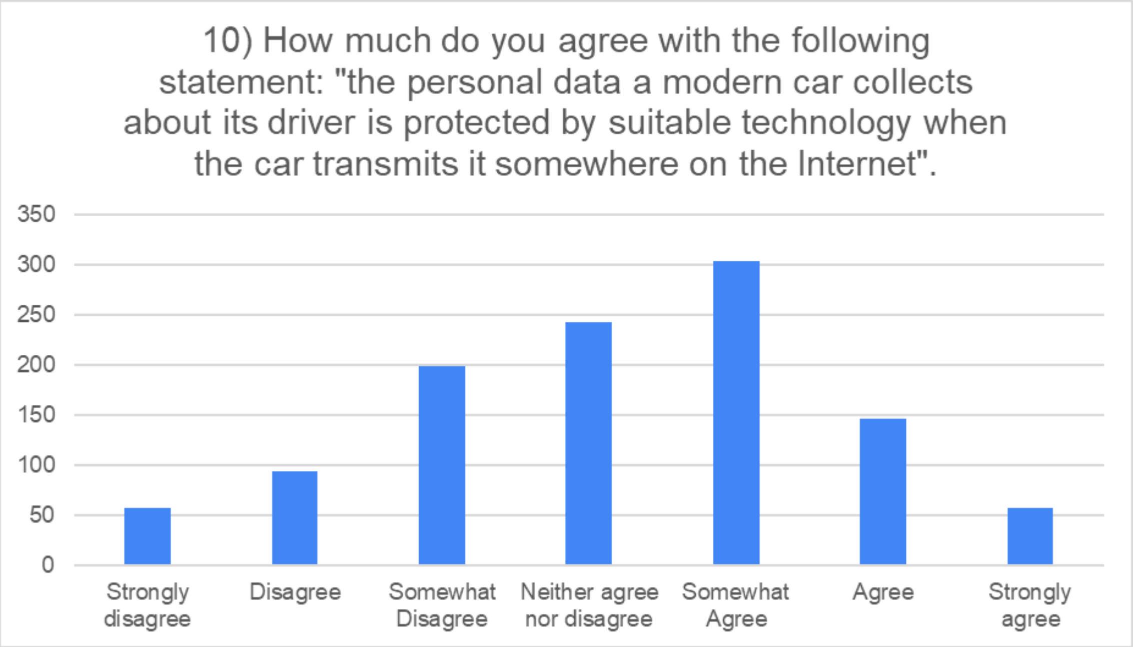


Figure A.11: Results question 10 (N = 1101)

Appendix B

Other Publications

In the following, it is reported a list of works published during my Ph.D. but not directly related to this thesis.

International Journals:

- Giampaolo Bella, Pietro Biondi, Stefano Bognanni. “Multi-service Threats: Attacking and Protecting Network Printers and VoIP Phones alike”. *Journal of Internet of Things.* Elsevier. 2022 [13].

International Conferences:

- Pietro Biondi, Stefano Bognanni, Giampaolo Bella. “Vulnerability Assessment and Penetration Testing on IP camera”. In: *International Conference on Internet of Things: Systems, Management and Security (IOTSMS 2021).* IEEE. 2021, pp. 1-8 [26].

- Pietro Biondi, Stefano Bognanni, Giampaolo Bella. “VoIP Can Still Be Exploited — Badly”. In: *International Conference on Fog and Mobile Edge Computing (FMEC).* IEEE. 2020, pp. 237-243 [25].

- Giampaolo Bella, Pietro Biondi. “You overtrust your printer”. In: *International Conference on Computer Safety, Reliability, and Security (SAFECOMP)*. LNCS, 2019, volume 11699. Pages 264-274 [12].

- Giuseppe Parasiliti, Pietro Biondi, Giuseppe Sgroi, Marzio Pennisi, Giulia Russo, Francesco Pappalardo. “A MapReduce tool for in-depth analysis of KEGG pathways: identification and visualization of therapeutic target candidates”. In *International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE. 2019 [103].

Bibliography

[1] Robert Abel. *Audi airbags disabled through software exploit*. https://www.crn.com.au/news/audi-airbags-disabled-through-software-exploit-410992. 2015.

[2] Z. O. Abu-Faraj et al. "Design and Development of a Heart-Attack Detection Steering Wheel". In: *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. 2018, pp. 1–6. DOI: 10.1109/CISP-BMEI.2018.8633210.

[3] Embedded Systems Academy. *CANcrypt*. https://www.cancrypt.eu/. 2018.

[4] ACKO. *Connected Cars: What is it? Features and Benefits*. https://www.acko.com/car-guide/connected-cars-features-benefits/. 2020.

[5] Ionut Arghire. *Researchers Find Exploitable Bugs in Mercedes-Benz Cars*. https://www.securityweek.com/researchers-find-exploitable-bugs-mercedes-benz-cars. 2021.

[6] Arm. *TrustZone for Cortex-M*. https://www.arm.com/technologies/trustzone-for-cortex-m. 2022.

[7] AUTOSAR. *Layered Software Architecture*. https://www.autosar.org/fileadmin/user_upload/standards/classic/19-11/AUTOSAR_EXP_LayeredSoftwareArchitecture.pdf. 2019.

[8] AUTOSAR. *Specification of Key Manager*. https://www.autosar.org/fileadmin/user_upload/standards/classic/19-11/AUTOSAR_SWS_KeyManager.pdf. 2019.

[9] AUTOSAR. *Specification of Secure Onboard Communication AUTOSAR CP R19-11*. https://www.autosar.org/fileadmin/user_upload/standards/classic/19-11/AUTOSAR_SWS_SecureOnboardCommunication.pdf. 2019.

[10] Ray Beaulieu et al. *The SIMON and SPECK Families of Lightweight Block Ciphers*. Cryptology ePrint Archive, Report 2013/404. https://eprint.iacr.org/2013/404. 2013.

[11] Henk Bekker. *Q1/2019 Europe: Best-Selling Car Manufacturers and Brands*. https://www.best-selling-cars.com/europe/q1-2019-europe-best-selling-car-manufacturers-and-brands/. 2019.

[12] Giampaolo Bella and Pietro Biondi. "You Overtrust Your Printer". In: *Computer Safety, Reliability, and Security (SAFECOMP)*. Ed. by Alexander Romanovsky et al. Cham: Springer International Publishing, 2019, pp. 264–274. ISBN: 978-3-030-26250-1.

[13] Giampaolo Bella, Pietro Biondi, and Stefano Bognanni. "Multi-service threats: Attacking and protecting network printers and VoIP phones alike". In: *Internet of Things* (2022), p. 100507. ISSN: 2542-6605. DOI: https://doi.org/10.1016/j.iot.2022.100507. URL: https://www.sciencedirect.com/science/article/pii/S2542660522000130.

[14] Giampaolo Bella, Pietro Biondi, and Giuseppe Tudisco. "A Double Assessment of Privacy Risks Aboard Top-Selling Cars". In: *Automotive Innovation* (2023), pp. 1–18. ISSN: 2096-4250. DOI: 10.1007/s42154-022-00203-2.

[15] Giampaolo Bella, Pietro Biondi, and Giuseppe Tudisco. “Car Drivers’ Privacy Concerns and Trust Perceptions”. In: *Trust, Privacy and Security in Digital Business*. Ed. by Simone Fischer-Hübner et al. Cham: Springer International Publishing, 2021, pp. 143–154. ISBN: 978-3-030-86586-3.

[16] Giampaolo Bella, Pietro Biondi, and Giuseppe Tudisco. “Car Drivers’ Privacy Concerns and Trust Perceptions”. In: *Trust, Privacy and Security in Digital Business*. Ed. by Simone Fischer-Hübner et al. Cham: Springer International Publishing, 2021, pp. 143–154. ISBN: 978-3-030-86586-3.

[17] Giampaolo Bella et al. “CINNAMON: A Module for AUTOSAR Secure Onboard Communication”. In: *2020 16th European Dependable Computing Conference (EDCC)*. 2020, pp. 103–110. DOI: 10.1109/EDCC51268.2020.00026.

[18] Giampaolo Bella et al. “Designing and implementing an AUTOSAR-based Basic Software Module for enhanced security”. In: *Computer Networks* (2022), p. 109377. ISSN: 1389-1286. DOI: https://doi.org/10.1016/j.comnet.2022.109377. URL: https://www.sciencedirect.com/science/article/pii/S138912862200411X.

[19] Giampaolo Bella et al. “Privacy and modern cars through a dual lens”. In: *2021 IEEE European Symposium on Security and Privacy Workshops (EuroS PW)*. 2021, pp. 136–143. DOI: 10.1109/EuroSPW54576.2021.00022.

[20] Giampaolo Bella et al. “Towards the COSCA framework for “COnseptualing Secure CArs”.” In: *Open Identity Summit 2021*. Ed. by Heiko Roßnagel, Christian H. Schunck, and Sebastian Mödersheim. Bonn: Gesellschaft für Informatik e.V., 2021, pp. 37–46.

[21] Giampaolo Bella et al. “Towards the COSCA framework for “COnseptualing Secure CArs”.” In: *Open Identity Summit 2021*. Ed. by Heiko Roßnagel, Christian H. Schunck, and Sebastian Mödersheim. Bonn: Gesellschaft für Informatik e.V., 2021, pp. 37–46.

[22] Cinzia Bernardeschi et al. “Modeling and Generation of Secure Component Communications in AUTOSAR”. In: *Proceedings of the Symposium on Applied Computing*. SAC ’17. Marrakech, Morocco: Association for Computing Machinery, 2017, pp. 1473–1480. ISBN: 9781450344869. DOI: 10.1145/3019612.3019682.

[23] Mario Luca Bernardi et al. “Driver and Path Detection through Time-Series Classification”. en. In: *Journal of Advanced Transportation* 2018 (2018), pp. 1–20. ISSN: 0197-6729, 2042-3195. DOI: 10.1155/2018/1758731. URL: https://www.hindawi.com/journals/jat/2018/1758731/ (visited on 03/25/2020).

[24] Lorenzo Franceschi Bicchierai. *Hacker Finds He Can Remotely Kill Car Engines After Breaking Into GPS Tracking Apps*. https://www.vice.com/en/article/zmpx4x/hacker-monitor-cars-kill-engine-gps-tracking-apps. 2019.

[25] Pietro Biondi, Stefano Bognanni, and Giampaolo Bella. “VoIP Can Still Be Exploited — Badly”. In: *2020 Fifth International Conference on Fog and Mobile Edge Computing (FMEC)*. 2020, pp. 237–243. DOI: 10.1109/FMEC49853.2020.9144875.

[26] Pietro Biondi, Stefano Bognanni, and Giampaolo Bella. "Vulnerability Assessment and Penetration Testing on IP camera". In: *2021 8th International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*. 2021, pp. 1–8. DOI: 10.1109/IOTSMS53705.2021.9704890.

[27] Alina Bizga. *Tesla Data Leak: Pre-Owned Vehicle Infotainment Components Store Owners' Personal Details and Passwords.* https://securityboulevard.com/2020/05/tesla-data-leak-pre-owned-vehicle-infotainment-components-store-owners-personal-details-and-passwords/. 2020.

[28] Brian Krzanich. *Data is the New Oil in the Future of Automated Driving.* https://newsroom.intel.com/editorials/krzanich-the-future-of-automated-driving/. 2016.

[29] Chris Brook. *Tesla Data Theft Case Illustrates the Danger of the Insider Threat.* https://digitalguardian.com/blog/tesla-data-theft-case-illustrates-danger-insider-threat. 2021.

[30] Alessandro Bruni et al. "Formal Security Analysis of the MaCAN Protocol". In: *Integrated Formal Methods*. Ed. by Elvira Albert and Emil Sekerinski. Cham: Springer International Publishing, 2014, pp. 241–255. ISBN: 978-3-319-10181-1.

[31] O.B. Chedzoy. *Phi Coefficient.* https://flowjo.typepad.com/the_daily_dongle/files/Phi-coefficient.pdf. 2006.

[32] Catalin Cimpanu. *Tesla car hacked at Pwn2Own contest.* https://www.zdnet.com/article/tesla-car-hacked-at-pwn2own-contest/. 2019.

[33] Catalin Cimpanu. *Toyota announces second security breach in the last five weeks*. https://www.zdnet.com/article/toyota-announces-second-security-breach-in-the-last-five-weeks/. 2019.

[34] CISOMAG. *Data Breach Affects 384,319 BMW Customers in the U.K.* https://cisomag.eccouncil.org/bmw-data-breach/. 2020.

[35] Kate Conger. *Apple will require HTTPS connections for iOS apps by the end of 2016*. https://techcrunch.com/2016/06/14/apple-will-require-https-connections-for-ios-apps-by-the-end-of-2016/. 2015.

[36] Lucian Constantin. *Researchers Hack Car Infotainment System and Find Sensitive User Data Inside*. https://www.vice.com/en/article/3kvw8y/researchers-hack-car-infotainment-system-and-find-sensitive-user-data-inside. 2017.

[37] Lucian Constantin. *Researchers hack Tesla Model S with remote attack*. https://www.pcworld.com/article/3121999/researchers-demonstrate-remote-attack-against-tesla-model-s.html. 2016.

[38] COSCA Team. *Assessment of car security risks and drivers' privacy risks*. https://cosca-project.dmi.unict.it/. 2020.

[39] COSCA Team. *Classifying data collected by cars*. https://cosca-project.dmi.unict.it/. 2020.

[40] Luca Dariz et al. "A Joint Safety and Security Analysis of message protection for CAN bus protocol". In: *Advances in Science, Technology and Engineering Systems Journal* 3.1 (2018), pp. 384–393. DOI: 10.25046/aj030147.

[41] Luca Dariz et al. "Trade-Off Analysis of Safety and Security in CAN bus communication". In: *The 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS 2017)*. Piscataway, New Jersey, USA: IEEE, 2017, pp. 226–231.

[42] Luca Dariz et al. "Trade-off analysis of safety and security in CAN bus communication". In: *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems, MT-ITS 2017, Naples, Italy, June 26-28, 2017*. IEEE, 2017, pp. 226–231. DOI: 10.1109/MTITS.2017.8005670. URL: https://doi.org/10.1109/MTITS.2017.8005670.

[43] Davey Winder. *Airbus, Porsche, Toshiba And Volkswagen Data Stolen In Massive Breach – What You Need To Know*. https://www.forbes.com/sites/daveywinder/2019/05/04/airbus-porsche-toshiba-and-volkswagen-data-stolen-in-massive-breach-what-you-need-to-know/. 2019.

[44] Sebastian Derikx, Mark de Reuver, and Maarten Kroesen. "Can privacy concerns for insurance of connected cars be compensated?" In: *Electronic Markets* 26.1 (Feb. 2016), pp. 73–81. ISSN: 1422-8890. DOI: 10.1007/s12525-015-0211-0. URL: https://doi.org/10.1007/s12525-015-0211-0.

[45] A. D. Dwivedi, P. Morawiecki, and G. Srivastava. "Differential Cryptanalysis of Round-Reduced SPECK Suitable for Internet of Things Devices". In: *IEEE Access* 7 (2019), pp. 16476–16486. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2019.2894337.

[46] M. Dworkin. *Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication*. NIST Special Publication 800-38B. May 2005.

[47] ENISA. *Risk Management - Principles and Inventories for Risk Management.* https://www.enisa.europa.eu/publications/risk-management-principles-and-inventories-for-risk-management-risk-assessment-methods-and-tools. 2006.

[48] European Commission. *eCall in all new cars from April 2018.* https://ec.europa.eu/digital-single-market/en/news/ecall-all-new-cars-april-2018. 2020.

[49] European Union. *General Data Protection Regulation (EU Regulation 2016/679).* https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=OJ:L:2016:119:FULL. 2016.

[50] European Union. *Guidelines 1/2020 on processing personal data in the context of connected vehicles and mobility related applications.* https://edpb.europa.eu/sites/edpb/files/consultation/edpb_guidelines_202001_connectedvehicles.pdf. 2020.

[51] Cyber Center of Excellence. *Data Leak Hits Nissan North America.* https://sdccoe.org/breach/data-leak-hits-nissan-north-america/. 2021.

[52] Adrienne Porter Felt and David Wagner. *Phishing on Mobile Devices.* https://people.eecs.berkeley.edu/~daw/papers/mobphish-w2sp11.pdf. 2011.

[53] Forbes. *Five Risk Assessments Where Automotive Giants Need To Improve To Compete With Startups.* https://www.forbes.com/sites/stevetengler/2021/07/08/five-risk-assessments-where-automotive-giants-need-to-improve-to-compete-with-startups/. 2021.

[54] Eclipse Foundation. *Eclipse Papyrus - Modeling environment.* https://www.eclipse.org/papyrus/. 2022.

[55] France24. *France's Renault hit in worldwide 'ransomware' cyber attack.* https://www.france24.com/en/20170512-cyberattack-ransomware-renault-worldwide-british-hospitals. 2017.

[56] Aurelien Francillon, Boris Danev, and Srdjan Capkun. *Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars.* https://eprint.iacr.org/2010/332.pdf. 2010.

[57] Umberto Fugiglando et al. *Characterizing the "Driver DNA" Through CAN Bus Data Analysis.* https://www.iit.cnr.it/sites/default/files/Driving_DNA.pdf. 2017.

[58] G. Costantino et al. "CANDY: A Social Engineering Attack to Leak Information from Infotainment System". In: *IEEE 87th VTC.* 2018. DOI: 10.1109/VTCSpring.2018.8417879.

[59] Scott Gayou. *Jailbreaking Subaru StarLink.* https://github.com/sgayou/subaru-starlink-research/blob/master/doc/README.md. 2018.

[60] Fabrizio Tronci Giampaolo Bella Pietro Biondi. "Papyrus-based safety analysis automatization". In: *6th International Conference on System Reliability and Safety (ICSRS), Venice, Italy.* Vol. In press. 2022.

[61] Naveen Goud. *Cyber Attack on Toyota Car Maker.* https://www.cybersecurity-insiders.com/cyber-attack-on-toyota-car-maker/. 2019.

[62] Andy Greenberg. *A New Wireless Hack Can Unlock 100 Million Volkswagens.* https://www.wired.com/2016/08/oh-good-new-hack-can-unlock-100-million-volkswagens/. 2016.

[63] Andy Greenberg. *Hackers Could Steal a Tesla Model S by Cloning Its Key Fob—Again.* https://www.wired.com./story/hackers-steal-tesla-model-s-key-fob-encryption/. 2019.

[64] Andy Greenberg. *Hackers Cut a Corvette's Brakes Via a Common Car Gadget.* https://www.wired.com/2015/08/hackers-cut-corvettes-brakes-via-common-car-gadget/. 2015.

[65] B. Groza and P. Murvay. "Security Solutions for the Controller Area Network: Bringing Authentication to In-Vehicle Networks". In: *IEEE Vehicular Technology Magazine* 13.1 (Mar. 2018), pp. 40–47. ISSN: 1556-6072. DOI: 10.1109/MVT.2017.2736344.

[66] Bogdan Groza et al. "Libra-can: a lightweight broadcast authentication protocol for controller area networks". In: *International Conference on Cryptology and Network Security*. Springer. Cham, 2012, pp. 185–200.

[67] Hackaday. *Reverse engineering the Renault Update List display - Part 1.* https://hackaday.io/project/27439-smart-car-radio/log/67874-reverse-engineering-the-renault-update-list-display-part-1. 2017.

[68] Ahmed Hazem and HA Fahmy. "Lcap-a lightweight can authentication protocol for securing in-vehicle networks". In: *10th escar Embedded Security in Cars Conference, Berlin, Germany*. Vol. 6. 2012.

[69] Will Houcheime. *Tesla Experiences Internal Breach, Leaking Valuable Company Data.* https://securityboulevard.com/2021/02/tesla-experiences-internal-breach-leaking-valuable-company-data/. 2021.

[70] Troy Hunt. *Controlling vehicle features of Nissan LEAFs across the globe via vulnerable APIs*. https://www.troyhunt.com/controlling-vehicle-features-of-nissan/. 2016.

[71] International Organization for Standardization. *ISO 26262-1: Road vehicles — Functional safety*. https://www.iso.org/standard/68383.html. 2018.

[72] International Organization for Standardization. *ISO/IEC 11889-1:2015 - Trusted Platform Module library*. https://www.iso.org/standard/66510.html. 2015.

[73] International Organization for Standardization. *ISO/IEC 27005:2018 Information technology — Security techniques — Information security risk management*. https://www.iso.org/standard/75281.html. 2018.

[74] International Organization for Standardization. *Road vehicles — Controller area network (CAN) — Part 1: Data link layer and physical signalling*. https://www.iso.org/standard/63648.html. 2015.

[75] IXXAT. *IXXAT Company*. https://www.ixxat.com/products/products-industrial/tools-overview/cananalyser. 2020.

[76] Jeff Crume. *OwnStar: Yet another car hack*. https://insideinternetsecurity.wordpress.com/2015/08/05/ownstar-yet-another-car-hack/. 2015.

[77] Jeremy Kirk. *Mercedes-Benz Data Leak Lesson: Lock Down Code Repositories*. https://www.bankinfosecurity.com/blogs/mercedes-benz-data-leak-embarrassing-but-endurable-p-2903. 2020.

[78] Howard Kass. *Mercedes-Benz hit by third-party data breach*. https://www.msspalert.com/cybersecurity-news/mercedes-benz-hit-by-third-party-data-breach/. 2021.

[79] John C. Knight. “Safety Critical Systems: Challenges and Directions”. In: *Proceedings of the 24th International Conference on Software Engineering*. ICSE ’02. Orlando, Florida: ACM, 2002, pp. 547–550. ISBN: 158113472X. DOI: 10.1145/581339.581406. URL: https://doi.org/10.1145/581339.581406.

[80] Eduard Kovacs. *Tesla Car Hacked Remotely From Drone via Zero-Click Exploit*. https://www.securityweek.com/tesla-car-hacked-remotely-drone-zero-click-exploit. 2021.

[81] Ryo Kurachi et al. “CaCAN-centralized authentication system in CAN (controller area network)”. In: *14th Int. Conf. on Embedded Security in Cars (ESCAR 2014)*. 2014.

[82] Mikhail Kuzin. *Mobile apps and stealing a connected car*. https://securelist.com/mobile-apps-and-stealing-a-connected-car/77576/. 2017.

[83] European Union Law. *UN Regulation No 155 – Uniform provisions concerning the approval of vehicles with regards to cybersecurity and cybersecurity management system*. https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:42021X0387&from=EN. 2021.

[84] Lee Johnstone. *Automotive Giant Audi Hacked, 2000+ Account Credentials leaked by Xc0unt3r*. https://www.databreaches.net/automotive-giant-audi-hacked-2000-account-credentials-leaked-by-xc0unt3r/. 2013.

[85] Lee Johnstone. *Peugeot Canada Hacked, Accounts and Data leaked by @Ag3nt47*. https://www.databreaches.net/peugeot-canada-hacked-accounts-and-data-leaked-by-ag3nt47/. 2013.

[86] Georg Macher et al. “Threat and Risk Assessment Methodologies in the Automotive Domain”. In: *Procedia Computer Science* (2016). DOI: 10.1016/j.procs.2016.04.268.

[87] Microsoft. *The STRIDE Threat Model*. https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20). 2009.

[88] Microsoft. *The STRIDE Threat Model*. https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20). 2020.

[89] Charlie Miller and Chris Valasek. “A survey of remote automotive attack surfaces”. In: *Black Hat USA* (2014).

[90] Michael Mimoso. *Unnamed, Popular ICS Firmware Contains Hard-Coded FTP Credential*. https://threatpost.com/unnamed-popular-ics-firmware-contains-hard-coded-ftp-credential/100941/. 2013.

[91] Miro Enev, Alex Takakuwa, Karl Koscher, and Tadayoshi Kohno. *Automobile Driver Fingerprinting*. https://petsymposium.org/2016/files/papers/Automobile_Driver_Fingerprinting.pdf. 2016.

[92] Felix Mohrle et al. “Automated compositional safety analysis using component fault trees”. In: *2015 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*. 2015, pp. 152–159. DOI: 10.1109/ISSREW.2015.7392061.

[93] Jean-Philippe Monteuuis et al. “SARA: Security Automotive Risk Analysis Method”. In: *Proceedings of the 4th ACM Workshop on Cyber-Physical System Security*. CPSS ’18. Incheon, Republic of Korea: Association for Computing Machinery, 2018, pp. 3–14. ISBN: 9781450357555. DOI: 10.1145/3198458.3198465.

[94] Motorbiscuit. *Major Software Flaw Leaves Ford and Volkswagen Cars Vulnerable to Hackers*. https://www.motorbiscuit.com/major-software-flaw-leaves-ford-volkswagen-cars-vulnerable-hackers/. 2021.

[95] Nicky Mouha et al. "Chaskey: An Efficient MAC Algorithm for 32-bit Microcontrollers". In: *Selected Areas in Cryptography – SAC 2014*. Ed. by Antoine Joux and Amr Youssef. Cham: Springer International Publishing, 2014, pp. 306–323.

[96] N. Du and J. Haspiel et al. "Look who's talking now: Implications of AV's explanations on driver's trust, AV preference, anxiety and mental workload". In: *Transportation Research Part C: Emerging Technologies* 104 (2019), pp. 428–442. ISSN: 0968-090X. DOI: 10.1016/j.trc.2019.05.025. URL: http://www.sciencedirect.com/science/article/pii/S0968090X18313640.

[97] National Association of Insurance Commissioners. *TELEMATICS/USAGE-BASED INSURANCE*. https://content.naic.org/cipr_topics/topic_telematicsusage_based_insurance.htm. 2020.

[98] Simona Negru. *Connected cars and in-car payments: the road so far and the road ahead*. https://tinyurl.com/Simona-Negru-cars. 2019.

[99] O.Hartkopp, C. Reuber, and R.Schilling. "MaCAN message authenticated CAN". In: ed. by Proc. 10th Int. Conf. Embedded Security in Cars (ESCAR). 2012.

[100] Lindsey O'Donnell. *Volkswagen Cars Open To Remote Hacking, Researchers Warn*. https://threatpost.com/volkswagen-cars-open-to-remote-hacking-researchers-warn/131571/. 2018.

[101] Charlie Osborne. *Over a dozen vulnerabilities uncovered in BMW vehicles.* https://www.zdnet.com/article/over-a-dozen-vulnerabilities-uncovered-in-bmw-vehicles/. 2018.

[102] Charlie Osborne. *Volkswagen, Audi disclose data breach impacting over 3.3 million customers, interested buyers.* https://www.zdnet.com/article/volkswagen-audi-disclose-data-breach-impacting-over-3-3-million-customers-interested-buyers/. 2021.

[103] Giuseppe Alessandro Parasiliti Palumbo et al. "A MapReduce tool for in-depth analysis of KEGG pathways: identification and visualization of therapeutic target candidates". In: *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM).* 2019, pp. 2157–2162. DOI: 10.1109/BIBM47256.2019.8982978.

[104] Petersen, Luke and Robert, Lionel and Yang, Xi Jessie and Tilbury, Dawn. *Situational Awareness, Driver's Trust in Automated Driving Systems and Secondary Task Performance.* May 2019.

[105] Andreea-Ina Radu and Flavio D. Garcia. "LeiA: A Lightweight Authentication Protocol for CAN". In: *Computer Security – ESORICS 2016.* Ed. by Ioannis Askoxylakis et al. Cham: Springer International Publishing, 2016, pp. 283–300. ISBN: 978-3-319-45741-3.

[106] Teri Robinson. *BMW customer database for sale on dark web.* https://www.scmagazine.com/home/security-news/bmw-customer-database-for-sale-on-dark-web/. 2020.

[107] Marc Ruef. *Car Hacking - Analysis of the Mercedes Connected Vehicle API.* https://www.scip.ch/en/?labs.20180405. 2018.

[108] Michal Sajdak. *The new hack allows wireless opening of over 100 million cars: Audi, Skoda, various VW, Ford, Citroen.* https://research.securitum.com/the-new-hack-allows-wireless-opening-of-over-100-million-cars-audi-skoda-various-vw-ford-citroen/. 2016.

[109] Mark Sauerwald. *CAN bus, Ethernet, or FPD-Link: Which is best for automotive communications?* https://www.ti.com/lit/an/slyt560/slyt560.pdf?ts=1590392933727. 2014.

[110] Brandon Schoettle and Michael Sivak. "A survey of public opinion about connected vehicles in the U.S., the U.K., and Australia". In: *2014 International Conference on Connected Vehicles and Expo (ICCVE)*. IEEE, Nov. 2014, pp. 687–692. DOI: 10.1109/ICCVE.2014.7297637.

[111] S. Schreiber et al. "UML-based safety analysis of distributed automation systems". In: *2007 IEEE Conference on Emerging Technologies and Factory Automation (EFTA 2007)*. 2007, pp. 1069–1075. DOI: 10.1109/EFTA.2007.4416901.

[112] Shane McGlaun. *Ford Data Breach Not An Issue Says Automaker.* https://fordauthority.com/2019/06/ford-data-breach-not-an-issue-says-automaker/. 2019.

[113] Ax Sharma. *Ford bug exposed customer and employee records from internal systems.* https://www.bleepingcomputer.com/news/security/ford-bug-exposed-customer-and-employee-records-from-internal-systems/. 2021.

[114] Smishad Thomas. *Automotive Risk Assessment with ISO 26262*. https://www.einfochips.com/blog/automotive-risk-assessment-with-iso-26262/. 2021.

[115] D. Stabili, L. Ferretti, and M. Marchetti. “Analyses of Secure Automotive Communication Protocols and Their Impact on Vehicles Life-Cycle”. In: *2018 IEEE International Conference on Smart Computing (SMARTCOMP)*. June 2018, pp. 452–457. DOI: 10.1109/SMARTCOMP.2018.00045.

[116] SysML Org. *SysML Open Source Project*. https://sysml.org/. 2022.

[117] Tencent Keen Security Lab. *Experimental Security Assessment on Lexus Cars*. https://keenlab.tencent.com/en/2020/03/30/Tencent-Keen-Security-Lab-Experimental-Security-Assessment-on-Lexus-Cars/. 2020.

[118] Tencent Keen Security Lab. *New Vehicle Security Research by KeenLab: Experimental Security Assessment of BMW Cars*. https://keenlab.tencent.com/en/2018/05/22/New-CarHacking-Research-by-KeenLab-Experimental-Security-Assessment-of-BMW-Cars/. 2018.

[119] The Institution of Engineering and Technology. *Serious cyber-security flaws uncovered in Ford and Volkswagen cars*. https://eandt.theiet.org/content/articles/2020/04/serious-cyber-security-flaws-uncovered-in-ford-and-volkswagen-cars-that-could-endanger-drivers/. 2020.

[120] Iain Thomson. *Stop the music! Booby-trapped song carjacked vehicles – security prof*. https://www.theregister.com/2016/01/26/hackers_can_take_full_control_of_car_os/. 2016.

[121] A. Toola. “The safety of process automation”. In: *Automatica* 29.2 (1993), pp. 541–548. ISSN: 0005-1098. DOI: https://doi.org/10.1016/0005-1098(93)90154-L. URL: https://www.sciencedirect.com/science/article/pii/000510989390154L.

[122] Jay Turla. *Mazda Infotainment USB Port PoC Attacks*. https://github.com/shipcod3/mazda_getInfo. 2017.

[123] Chris Valasek and Charlie Miller. *Remote Exploitation of an Unaltered Passenger Vehicle*. http://illmatics.com/Remote%20Car%20Hacking.pdf. 2015.

[124] Anthony Van Herrewege, Dave Singelee, and Ingrid Verbauwhede. “CANAuth-a simple, backward compatible broadcast authentication protocol for CAN bus”. In: *ECRYPT Workshop on Lightweight Cryptography. Vol. 2011*. 2011, pp. 1–7.

[125] Sebastian Vasile, David Oswald, and Tom Chothia. *Breaking all the Things — A Systematic Survey of Firmware Extraction Techniques for IoT Devices*. https://www.cs.bham.ac.uk/~tpc/Papers/CARDIS18.pdf. 2019.

[126] Serge Vaudenay. “Security Flaws Induced by CBC Padding — Applications to SSL, IPSEC, WTLS...” In: *Advances in Cryptology — EUROCRYPT 2002*. Ed. by Lars R. Knudsen. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 534–545. ISBN: 978-3-540-46035-0.

[127] Yunpeng Wang et al. “A Systematic Risk Assessment Framework of Automotive Cybersecurity”. In: *Automotive Innovation* 4.3 (Aug. 2021), pp. 253–261. ISSN: 2522-8765. DOI: 10.1007/s42154-021-00140-6. URL: https://doi.org/10.1007/s42154-021-00140-6.

[128] Zack Whittaker. *Flaws in third-party software exposed dozens of Teslas to remote access*. https://techcrunch.com/2022/01/24/teslamate-bug-teslas-exposed-remote/. 2022.

[129] Zack Whittaker. *Security flaws let anyone snoop on Guardzilla smart camera video recordings*. https://techcrunch.com/2018/12/27/guardzilla-security-camera-flaws/. 2018.

[130] Wikipedia. *Birthday Attack*. https://en.wikipedia.org/wiki/Birthday_attack. 2021.

[131] Wikipedia. *CSMA/BA*. https://it.wikipedia.org/wiki/CSMA/BA. 2021.

[132] Wikipedia. *Failure mode and effects analysis*. https://en.wikipedia.org/wiki/Failure_mode_and_effects_analysis. 2021.

[133] Wikipedia. *Fault tree analysis*. https://en.wikipedia.org/wiki/Fault_tree_analysis. 2021.

[134] Wikipedia. *P-Value*. https://en.wikipedia.org/wiki/P-value. 2020.

[135] Wikipedia. *P-value*. https://en.wikipedia.org/wiki/P-value. 2022.

[136] Wikipedia. *Pearson correlation coefficient*. https://en.wikipedia.org/wiki/Pearson_correlation_coefficient. 2022.

[137] Wikipedia. *Pearson Product-Moment Correlation*. https://en.wikipedia.org/wiki/Pearson_correlation_coefficient. 2020.

[138] Wikipedia. *Phi coefficient*. https://en.wikipedia.org/wiki/Phi_coefficient. 2020.

[139] Wikipedia. *Point-biserial correlation coefficient*. https://en.wikipedia.org/wiki/Point-biserial_correlation_coefficient. 2022.

[140] Wikipedia. *Spearman's rank correlation coefficient.* https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient. 2020.

[141] Wikipedia. *Spearman's rank correlation coefficient.* https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient. 2022.

[142] Wikipedia. *Technology readiness level.* https://en.wikipedia.org/wiki/Technology_readiness_level. 2022.

[143] Wikipedia. *Unified Modeling Language.* https://en.wikipedia.org/wiki/Unified_Modeling_Language. 2022.

[144] Marko Wolf and Michael Scheibel. "A systematic approach to a qualified security risk analysis for vehicular IT systems". In: *Automotive - Safety & Security 2012.* Ed. by Erhard Plödereder et al. Bonn: Gesellschaft für Informatik e.V., 2012, pp. 195–210.

[145] Xuhang Ying et al. "TACAN: Transmitter Authentication through Covert Channels in Controller Area Networks". In: *CoRR* abs/1903.05231 (2019). arXiv: 1903.05231. URL: http://arxiv.org/abs/1903.05231.

[146] Artem Yushev et al. "TLS-over-CAN: An Experimental Study of Internet-Grade End-to-End Communication Security for CAN Networks". In: *IFAC-PapersOnLine* 51.6 (2018). 15th IFAC Conference on Programmable Devices and Embedded Systems PDeS 2018, pp. 96–101. ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2018.07.136. URL: http://www.sciencedirect.com/science/article/pii/S2405896318308802.

[147] Zack Whittaker. *Mercedes-Benz app glitch exposed car owners' information to other users.* https://techcrunch.com/2019/10/19/mercedes-benz-app-glitch-exposed/. 2019.

[148] Kexiong (Curtis) Zeng et al. *All Your GPS Are Belong To Us: Towards Stealthy Manipulation of Road Navigation Systems*. https://people.cs.vt.edu/gangwang/sec18-gps.pdf. 2018.

[149] Lin Zhao et al. "Failure Propagation Modeling and Analysis via System Interfaces". In: *Mathematical Problems in Engineering* 2016 (May 2016), p. 8593612. ISSN: 1024-123X. DOI: 10.1155/2016/8593612.

[150] Tobias Ziermann, Stefan Wildermann, and Jürgen Teich. "CAN+: A new backward-compatible Controller Area Network (CAN) protocol with up to 16x higher data rates". In: *Proceedings of the Conference on Design, Automation and Test in Europe*. European Design and Automation Association. 2009, pp. 1088–1093.