IT-Security

Securing Electronic Control Units against emerging vehicle technology threats

Masterarbeit

zur Erlangung des akademischen Grades

Diplom-Ingenieur/in

eingereicht von

Fabian Hirnschal

is161515

im Rahmen des
Studienganges IT-Security an der Fachhochschule St. Pölten

Betreuung
Betreuer/in: FH-Prof. Mag. Dr. Simon Tjoa
Mitwirkung:

St. Pölten, January 23, 2022 _____ _____

(Unterschrift Verfasser/in) (Unterschrift Betreuer/in)

*

Ehrenwörtliche Erklärung

Ich versichere, dass

- ich diese Diplomarbeit selbständig verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und mich sonst keiner unerlaubten Hilfe bedient habe.
- ich dieses Diplomarbeitsthema bisher weder im Inland noch im Ausland einem Begutachter/einer Begutachterin zur Beurteilung oder in irgendeiner Form als Prüfungsarbeit vorgelegt habe.
- diese Arbeit mit der vom Begutachter/von der Begutachterin beurteilten Arbeit übereinstimmt.

Der Studierende/Absolvent räumt der FH St. Pölten das Recht ein, die Diplomarbeit für Lehre- und Forschungstätigkeiten zu verwenden und damit zu werben (z.B. bei der Projektevernissage, in Publikationen, auf der Homepage), wobei der Absolvent als Urheber zu nennen ist. Jegliche kommerzielle Verwertung/Nutzung bedarf einer weiteren Vereinbarung zwischen dem Studierenden/Absolventen und der FH St. Pölten.

Ort, Datum *Unterschrift*

Kurzfassung

Rasante Entwicklungen in der Fahrzeugindustrie verändern die Automobillandschaft und werden noch weitere Innovationen hervorbringen. Anwendungen für vernetzte Fahrzeuge sowie neue Funktionalitäten und Schnittstellen bringen neue Innovationen hervor doch ziehen auch Sicherheitsrisiken mit sich. Fahrzeughersteller benötigen angebrachte Sicherheitsmaßnahmen, um einen sicheren Fahrbetrieb gewährleisten zu können da Forscher bereits 2013 nachweisen konnten, dass Sicherheitslücken bei Dienstleistern, Carsharing-Unternehmen, Apps und autointernen Komponenten zu gefährlichen bis hin zu lebensbedrohlichen Situationen führen können.

Moderne Fahrzeugfunktionen werden durch komplexe vernetzte eingebettete Systeme, auch bekannt als elektronische Steuergeräte (ECUs) umgesetzt, die als Grundlage für einen sicheren Fahrzeugbetrieb angesehen werden können. Daher ist es für die Fahrzeugindustrie von entscheidender Bedeutung, die Integrität und beabsichtigte Funktionalität dieser Geräte sicherzustellen.

Diese Arbeit analysiert neue Fahrzeugtechnologien und präsentiert eine Bedrohungs- und Risikoanalyse unter Verwendung einer Methodik, die in der kürzlich veröffentlichten Norm ISO/SAE 21434 zur Cybersicherheit in Straßenfahrzeugen vorgeschlagen wird um Bedrohungen, Risiken und Sicherheitsziele zum Schutz von diesen Steuergeräten vor Cyberangriffen zu gewährleisten. Abschließend werden technische Sicherheitsmaßnahmen vorgeschlagen, um Steuergeräte vor Sicherheitsbedrohungen in der Automobillandschaft zu schützen.

Abstract

Developments in vehicle technologies are rapidly rapidly reshaping the automotive landscape and are expected to further evolve. While connected car applications introduce new features and interfaces to satisfy customers, they also raise security risks which require manufacturers to implement security measures to ensure safe vehicle operation. Researchers have demonstrated, that as early as 2013, security vulnerabilities were found in service providers, car sharing companies, apps and car-internal components which can lead to dangerous and even life-threatening situations.

Modern vehicle functionalities are realized by complex interconnected embedded devices also known as electronic control units (ECUs), which can be considered as the foundation to ensure safe vehicle operation. Therefore it is of critical importance for the industry to protect the integrity and intended functionality of these devices.

This paper analyzes emerging vehicle technologies and presents a threat and risk analysis using a methodology proposed in the recently published ISO/SAE 21434 standard regarding cyber security in road vehicles to determine threats, risks and security goals to protect ECUs against cyber attacks. Finally, technical security measures are proposed to secure ECUs against cyber security threats in the automotive landscape.

Contents

1. Introduction

Rapid developments in digital technology are reshaping innovations in vehicles, production and new business models like on-demand mobility services.[1] Studies have also shown, that the connected vehicle market is currently growing quickly. [2] Vehicles are expected to evolve further in the future and new technical developments like IoT, telematics and smart platforms enable features like enhanced communications, connectivity and vehicle management for all traffic participants. Connectivity applications such as navigation, mobile applications and in-car entertainment systems introduce new features as well as interfaces to other systems outside of the vehicle. These developments satisfy customer expectations but also raise security risks and require safeguards by manufacturers to ensure proper operation and passenger and traffic participant safety. Researchers have demonstrated, that as early as 2013, security vulnerabilities were found in service providers, car sharing companies, apps and car-internal components. [3] Unlike the impact on traditional IT systems, the lack of security measures in vehicles can lead to dangerous and even life-threatening situations. [4]

Modern vehicles are complex systems with various interconnected embedded devices also known as electronic control units ECUs (2.1.1). These devices are connected through various communication channels and protocols which can be accessed by different interfaces, both wired and wireless. This evolution has drastically increased the attack surface of such devices and vehicles in general and has rendered previous safeguards like redundancy and reliability as insufficient. All these factors contribute to the need of enhanced security functionalities of these control units, which are also responsible for controlling the engine, breaks, gearbox and various other components. [5] The safety of modern vehicles can consequently only be ensured by establishing an appropriate level of security in these embedded devices.

1.1. Problem Description

In the beginning of vehicle development, cars were almost exclusively operated mechanically and security measures were restricted to mechanical keys, locks and alarm devices to prevent theft. [6] Today's vehicles include devices with up to 100 million lines of code and wireless connectivity technologies, turning the vehicle into an almost permanently connected device in the internet, thereby increasing the complexity and attack surface regarding cyber security. Studies have shown, that this trend will continue in the next years. [7]

The automotive industry is also becoming more digitized and Advanced Driver Assistance Systems (ADAS) are becoming basic customer requirements. Although research has shown that drivers have misconceptions about ADAS systems and their capabilities. Customers who use these features are more reliant on these technologies and security and safety risks arise due to their dependence on these technologies.[8] While people benefit from these features in their cars, they are also becoming potential targets of cyber attacks. [4]

As vehicles have transformed from mechanical devices to complex interconnected systems, the embedded computers used in today's automotive landscape are essential for all the complex features to function reliably and are therefore an attractive target for cyber attacks. Unlike IT systems, these embedded devices need to operate in real-time [9] and have limited resources which raises additional challenges when it comes to resource intensive cryptographic operations.

Whilst regulatory bodies are aware of these challenges and some standards for vehicle security like the "Uniform provisions concerning the approval of vehicles with regards to cyber security and cyber security management system" [10] published in a first revision by the United Nations in 2020 do exist, this particular regulation only provides a high level description of threats and relating vulnerabilities for the entire vehicle. Therefore an appropriate security level for individual control units needs to be developed by each OEM individually. The ASPICE standard on the other hand is a process assessment- and reference model regarding software development in the automotive industry and references security requirements and aspects, but does not go into further detail regarding the implementation of security measures. [11]

Therefore, the problems mentioned above demonstrate that there is a critical need for protecting each individual electronic control unit against security threats [12] originating from emerging vehicle technologies.

1.2. Proposed Solution

The goal of this paper is to determine specific security measures for ECUs to enable OEMs to address the security challenges that arise through emerging vehicle technologies.
During the course of this work, a literature analysis will be conducted determining current developments in the automotive sector, emerging vehicle technologies and security standards in the automotive landscape. The resulting features and findings will be considered in a threat and risk assessment for an individual electronic control unit. The identified risks shall be evaluated based on the recently published ISO SAE 21434 standard [13] risk assessment proposal and countermeasures shall be defined to determine concrete security measures for ECUs.

1.3. Objectives

Briefly outlined, the main contributions of this thesis are as follows:

- Emerging vehicle technology features are identified
- Challenges regarding cyber security of ECUs are outlined
- Threats to individual ECUs are identified
- Security goals to protect assets against threat scenarios are defined
- Security measures for individual ECUs to mitigate the identified threats are proposed

2. Background

This chapter will first of all touch on the background and underlying technologies to provide the required understanding to follow the work in this paper. Next, a literature analysis is conducted regarding technologies, features and functionalities of emerging vehicle technologies. Finally, an overview of related areas of work is provided.

2.1. Fundamentals and Terminology

This section shall provide the basic understanding of the technologies and terms that are vital for the understanding of this work.

2.1.1. Electronic Control Unit (ECU)

ECUs are embedded computers in vehicles to control the electronics of mechanical systems or components. [5] They take inputs from sensors and other ECUs, but also send signals and data to actuators and other components to control vehicular functionalities. As an example, the engine control unit is mainly responsible to control the engine in terms of injection and ignition. It takes inputs from power-train system sensors like engine status and position, but also processes this data to drive actuators to accelerate or decelerate based on the position of the acceleration pedal. [14] Besides the physical system architecture in a vehicle consisting of multiple ECUs, each individual ECU also has its own physical architecture consisting of three main parts:

Application Software: Software components responsible for executing vehicle functionalities provided by the ECU (e.g. cruise control)

Middleware Software: Software components that are responsible for providing services for the application software to use and can by considered as library functions when compared to traditional IT systems (e.g. data transmission/reception, diagnosis features)

Hardware: Components and their drivers responsible for controlling physical hardware units (e.g. processor, buses)

2.1.2. Hardware Security Module (HSM)

A hardware security module is a physical device acting as a trusted security anchor and provider for different systems. It is designed to provide security functionalities over secured interfaces and storage for data in a secured environment. HSMs usually consist of a processor, non-volatile memory, communication interface, internal random access memory and can also provide features such as true/pseudo random number generation or hardware accelerated cryptographic modules. [15] The core features of such a device are listed below:

- Protection of SW security measures by acting as a trusted anchor
- Securely store, generate and process cryptographic and security relevant material inaccessible from outside by software
- Restrict hardware tampering
- Accelerate security mechanisms through specialized cryptographic hardware

2.1.3. E/E-Architecture

High end vehicles are made up of 70 ECUs or more which are connected to many buses. Which buses are utilized depends on the application area. Some examples of theses buses are CAN, LIN, CAN-FD, MOST, FlexRay and Ethernet which are all connected through a central gateway and will be detailed further in the next subsection. [5] The central gateway as shown in figure 2.1 is responsible for converting data from one bus format to another and moreover directly handles inter-ECU communications. Therefore it has to deal with a huge load of communication data as all the ECUs pass through the gateway. The figure illustrates the architecture of a BMW 7 series vehicle. It contains different types of buses (K-CAN System, MOST, K-CAN Periphery, SI-BUS and PT-CAN) with multiple ECUs connected to each bus.

2.1.4. In-Vehicle network

As already briefly discussed in the E/E-Architecture subsection, the In-Vehicle network (IVN) consists of all the ECUs, sensors, actuators and infotainment systems to enable the communication between relevant nodes. Historically, the IVN has been a closed network bus without external connection except for diagnosis purposes. [16] Due to the growing demand for connected services and ADAS, several different types of networks and bus protocols were developed to address different requirements within the vehicle. Table 2.1 provides an overview of different technologies and their typical use cases which will be discussed in this section.

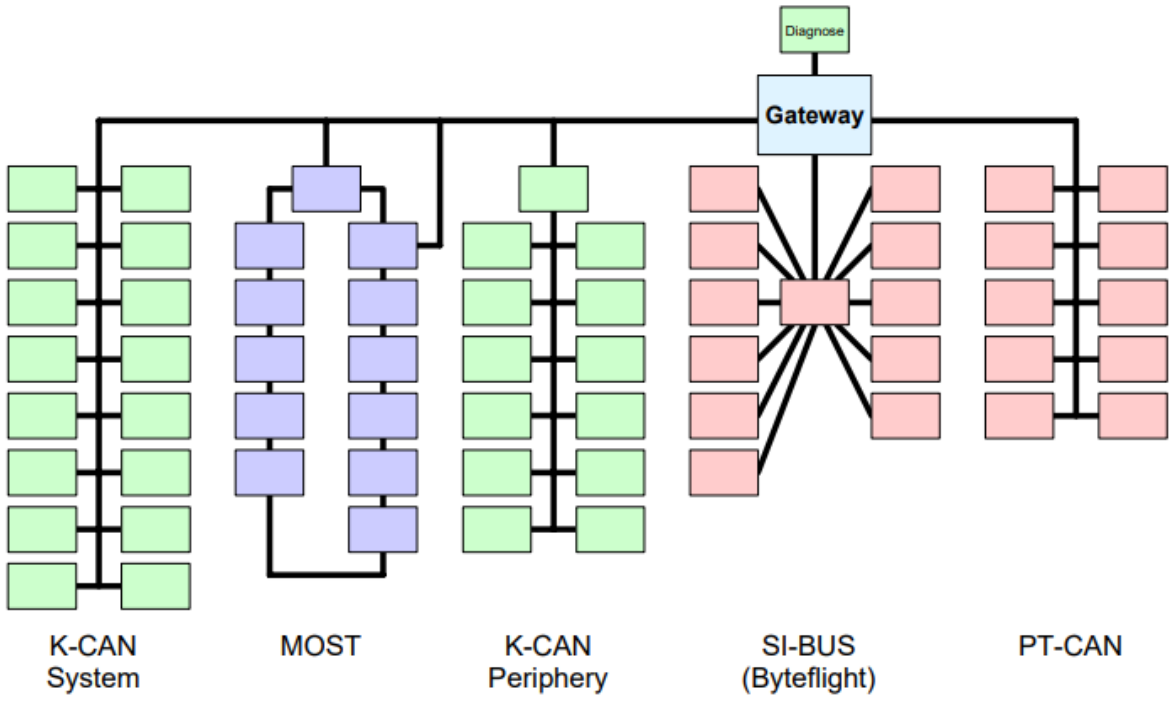

Figure 2.1.: Example In-vehicle E/E-architecture of BMW 7 series [5]

2.1.5. Controller Area Network (CAN)

The Controller area Network is a International Standardization Organization (ISO) defined serial communication protocol with an application from high speed networks to low cost multiplex wiring. It has a high immunity to electrical interference and a self-diagnose and data error repair ability. [18] The protocol has been widely applied in automotive communication and is also used in other sectors. Compared to TCP/IP it does not utilize source and destination addresses but uses a broadcasting communication technique in a way, that all nodes can send and receive packets on the bus. [19] Such an architecture is depicted in figure 2.2.

CAN FD (Flexible Data rate) is the successor of CAN as additional bandwidth requirements in the vehicle domain rendered the traditional CAN bus obsolete. CAN FD provides a higher bandwidth by allowing payloads of up to 64 byte with an increased bit rate.

2.1.6. Local Interconnect Network (LIN)

The Local Interconnect Network was developed to supplement CAN in cost critical applications where data transfer requirements are low. It is a single master serial bus which enables effective communica-

[1] X-by-Wire is a generic term referring to the replacement of mechanical or hydraulic systems, such as braking or steering, by electronic ones. [17]

SAE class	Bitrate [kbit/s]	typical technology	typical use case
A	<25	LIN	sensors / actuators
B	25 to 125	CAN (Low Speed)	ECUs e.g. comfort systems
C	125 to 1000	CAN (High Speed)	ECUs e.g. powertrain
C+	>= 1000	FlexRay	X-by-Wire[1]
D	>= 10 000	MOST, Ethernet	Multimedia

Table 2.1.: SAE classification of In-Vehicle network technologies

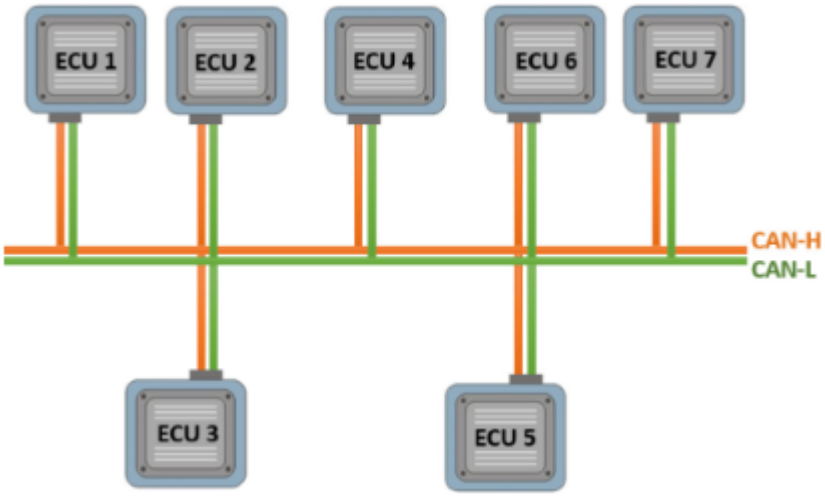

Figure 2.2.: CAN architecture [20]

tion in situations where the versatility and bandwidth of CAN is not required. Typical applications are functionalities like climate regulation, mirror-, door- or window control.[21] An example architecture of LIN is illustrated in figure 2.3.

2.1.7. Media Oriented System Transport (MOST)

These protocols, managed by the MOST Cooperation, are used for connecting multiple devices responsible for infotainment including navigation, digital radios, phones, displays, etc. It is a serials communication protocol following a ring topology to connect the nodes shown in figure 2.4. The plug and play nature of the protocol and its physical transmission medium with good electromagnetic compatibility properties [22] makes it preferable for multimedia applications.

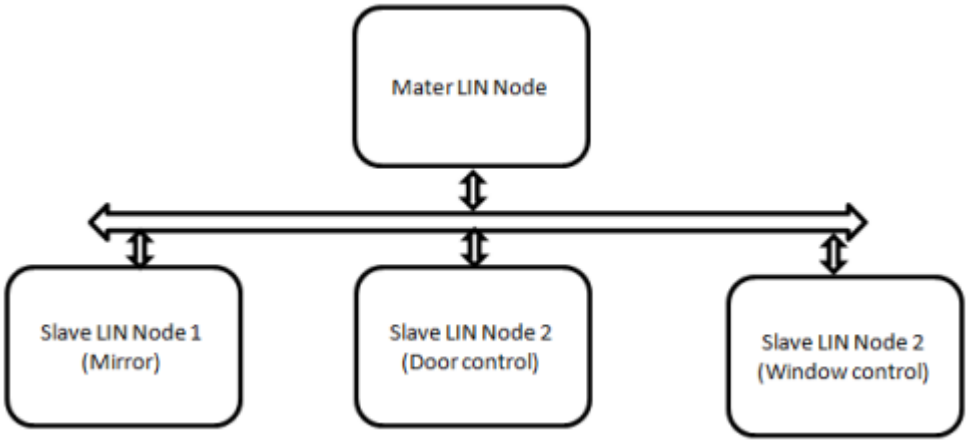

Figure 2.3.: LIN architecture [21]

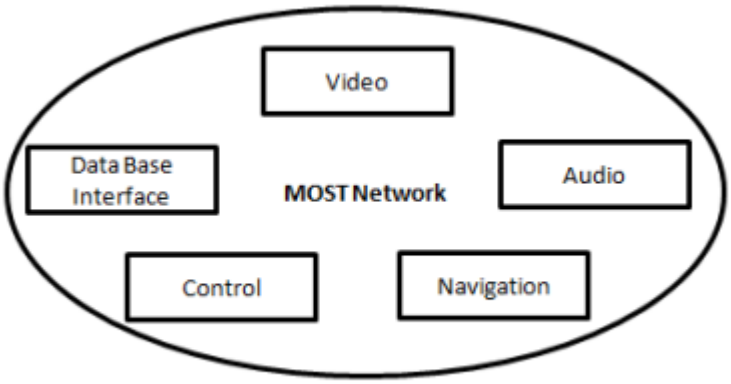

Figure 2.4.: MOST network [21]

2.1.8. FlexRay

The FlexRax protocol is considered to be the next step of CAN and LIN bus systems and designed to be more reliable and faster. It supports different topologies like bus, star and hybrid so it can be considered more versatile. Nevertheless the arbitration mechanism used by CAN and LIN is not used in FlexRay, which addresses the issue of data collision in a different manner. All nodes are synchronized to the common clock using Time Division Multiple Access (TDMA) which makes it less suitable for hard real-time applications than CAN. [20] Table 2.2 shows a comparison between CAN and FlexRay regarding their application domain, message transmission and data rate. [23]

Attributes	Bus System	
	CAN-Bus	*FlexRay*
Type of Real-time System	Hard real-time	Firm Real-time
Used in	Non-Deterministic Data	Deterministic Data
Message Transmission	Asynchronous	Synchronous&Asynchronous
Data Rate	1 Mbps	10 Mbps

Table 2.2.: CAN and FlexRay comparison [20]

2.1.9. Automotive Ethernet

Various Ethernet standards have been developed across different physical mediums like copper or fiber optic cables with speeds in the range of hundreds of Gbit/s. But due to different requirements between regular IT systems and the automotive landscape, new standards have been developed to tackle problems like interference and real-time applications. [23]

The development of T1-transmission methods (100Base-T1 and 1000BASE-T1) are of critical importance to automotive Ethernet as they allow the usage of Ethernet with price-competitive and easy to install wiring. Without the usage of Shielded Twisted Pair (STP) wiring, there would have been no cost benefit compared to the MOST technology which would have driven the industry to use the MOST150 standard. Also the future bandwidth potential, good scalability, reduced connectivity costs and compatability advantages for connecting vehicles to broader smart-infrastructure are driving the adoption of Automotive Ethernet. It is currently mainly used in diagnostic and infotainment applications but considered the 'bus of the future' by providing the primary interconnect for modern vehicles. [24]

2.1.10. Automotive Open System Architecture (AUTOSAR)

Modern vehicles can consist of up to 100 ECUs from different vendors and suppliers which need to work flawlessly with each other to support complex vehicular functionalities. Before the AUTOSAR standard was introduced, there was no standardized platform which could be used by OEMs and their suppliers to have a unified design of ECU software. Whenever an OEM wanted to change a component or supplier, this transition was therefore incredibly difficult as the new component needed to be integrated into ongoing developments. To solve this problem, reduce development time and costs and improve software quality, suppliers, manufacturers and OEMs created a consortium called the Automotive Open System

Architecture (AUTOSAR). [25] The standard was introduced as a joint partnership in 2003 consisting of more than 150 global partners and is considered a de facto standard in the automotive domain. [26] Figure 2.5 shows the AUTOSAR classic platform in its Release R21-11. Its architecture distinguishes three software layers on its highest abstraction level which run on a micro-controller:

Application Layer is a software layer, mostly hardware independent and communicates between software components and accesses the BSW via the Runtime Environment (RTE)

Runtime Environment (RTE) represents the full interface for applications

Basic Software (BSW) provides services, ECU and micro-controller abstraction and is furthermore divided into functional groups representing the memory, infrastructure system, drivers and communication services

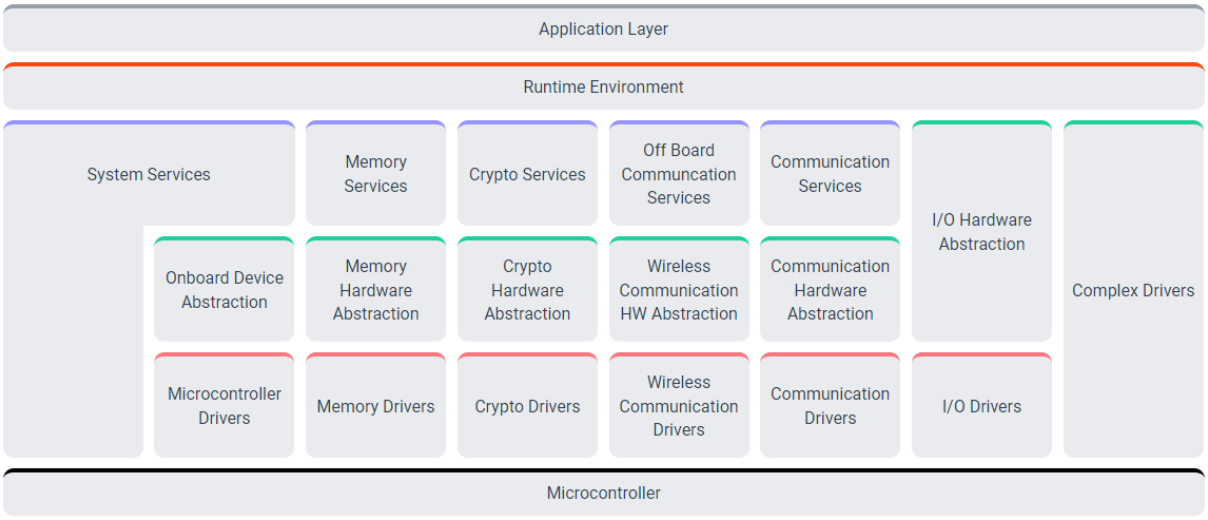

Figure 2.5.: AUTOSAR classic platform [27]

The lowest layer of the architecture (depicted in red) represents the Microcontroller Abstraction Layer, consisting of various drivers to enable the interoperability to the microcontroller. The Hardware Abstraction Layer in the middle of the BSW architecture (depicted in green) provides generic interfaces to the drivers and makes access independent of underlying hardware and drivers. The highest layer of the architecture (depicted in purple) offers standardized access to various services for applications via the RTE.

Crypto Stack

AUTOSAR already considers security functionalities in its architecture in the form of the Crypto Stack consisting of the Crypto Services, Crypto Hardware Abstraction and Crypto Drivers layer (see 2.5) ex-

panding over all layers in the AUTOSAR Layered Architecture. [28] It does not provide a security concept itself, but offers cryptographic services to the application which can be used to realize and support a certain security concept.

2.2. Emerging vehicle technologies

Technology and innovations in the automotive sector with the aim to make vehicles carbon free, available on-demand and driver-less are evolving fast. Available information suggests that there is significant potential in such technologies to improve road safety, lower fuel consumption and emissions, provide increased travel satisfaction and smart mobility solutions throughout the population. [29] However, the adoption speed of such technologies is still unclear as the automotive sector needs to overcome various technological, regulatory and cyber security challenges.
This section will cover a literature analysis and determine current innovations, developments and upcoming technologies in the automotive sector.

2.2.1. Advanced Driver Assistance Systems (ADAS)

Accident reducing systems have become crucial in the automotive industry and are becoming basic customer requirements. The first driver assistance system was the Anti-Lock Braking System (ABS) introduced in the late 1970s and can now be found in almost every modern vehicle. Studies have shown, that dynamic driving controls like ABS, the Traction Control System or Electronic Stability Control are the second most efficient safety system for passengers, right behind the seat belt. [30] Since November 2014 the ESC is even a legal requirement for new cars in the EU. Newly developed cars are tested today by the Euro NCAP, which has created a five-start safety rating system to help in comparing the safety mechanisms of vehicles more easily. Their safety rating is determined by a series of vehicle tests, designed and carried out by Euro NCAP. [31]
Many automobile companies are also performing research on driver inattention systems. These features are mainly based on vehicle based features and use different mechanisms to detect driver fatigue like eyelid activity, lane tracking or steering patterns. For example, the system used by Nissan adapts to the drivers behaviour and establishes a baseline whereas continuous statistical analysis of the steering corrections is conducted and if deviations from the baseline are detected a notification regarding detected fatigues is provided to the driver. [32] Advanced mechanisms of driving vigilance monitoring systems are being developed by car manufactures like Mercedes-Benz attention assist system which consists of sensors and cameras embedded in the cockpit, monitoring the drivers reaction, expression and face in

real-time. This data is then processed to evaluate the drivers emotions like stress, fatigue or even anger. [33]

ADAS is considered an advancement of driver assistant systems (DAS), which warns, informs, increases comfort or reduces workload by maneuvering or stabilizing the vehicle. As a subset of DAS, ADAS can be seen as advanced methods with increased use of complex algorithms to first detect and evaluate the environment of the vehicle based on collected data from a variety of sensor inputs. [34] Modern ADAS are controlling trajectories beyond the current driver request. These features and automated decisions require a high level of certainty which can only be achieved with an interconnected set of sensors. ADAS consist mainly of the following physical components: RADAR, LIDAR, cameras, night vision and ultrasonic sensors. These sensors allow vehicles to monitor both near and far fields in all directions and advances in technology with innovations such as improved image processing, better sensor fusion algorithms or positioning systems are continuously enabling manufacturers to provide new improvements when it comes to assistance systems.

Sensor fusion concepts are one of the key advancements in ADAS. This process refers to the internal processing by taking multiple inputs from a variety of external sensors and creating a map of possible areas of interest around the vehicle. This map will then be processed to compute a series of possible reactions or actions by analysing the situation. [35]

This situational analysis may be conducted by machine learning techniques in the future. Researchers such as Moujahid et. al [33] have shown, that some forms of machine learning may be used by car manufacturers in the future to predict or detect dangerous situations or ensure safe driving situations in different applications. They are relevant for preprocessing steps, during vehicle development but will also play a role in decision making in future vehicles. Machine learning methods are expected to become more and more related to ADAS and autonomous driving.

Table 2.3 provides a structured overview of different driver assistance features, which can be grouped into 6 different classifications.

Figure 2.6 shows the usage of ADAS sensors used in modern vehicles and provides an overview of their approximate range and individual use cases.

For a better understanding of the underlying technology of ADAS, the sensors and technologies utilized to provide such features are explained in further detail below:

Radio Detection and Ranging (RADAR)

Short-range radar (SRR) measures distances to an object up to 50m and is used in applications where the area close to the vehicle is affected, like Blind Spot Detection (BSD) and monitoring the area next to

	DAS	ADAS
Driving Stability	Differential, Antiblocking system, traction control, steering support, Electronic stability program, ...	-
Longitudinal Control	Cruise Control, Hill climb automatic, brake control	Adaptive cruise control, Speed adaption, Emergency break, Traffic sign/light violation warning, Speed limit information, Active Distance Assist
Lateral Control		Blind spot detection, Lane departure warning, Lane keeping assist, Lane change support
Cockpit	Fatigue monitoring, Passenger seat observation, Navigation	Indoor monitoring
Parking	Surround view, Park distance control	Active/Remote Parking Assist,
Light and Sight	Rain sensor, Adaptive light system	Headlight range adaption, Beam light assist

Table 2.3.: Driver assistance system features and classification [36]

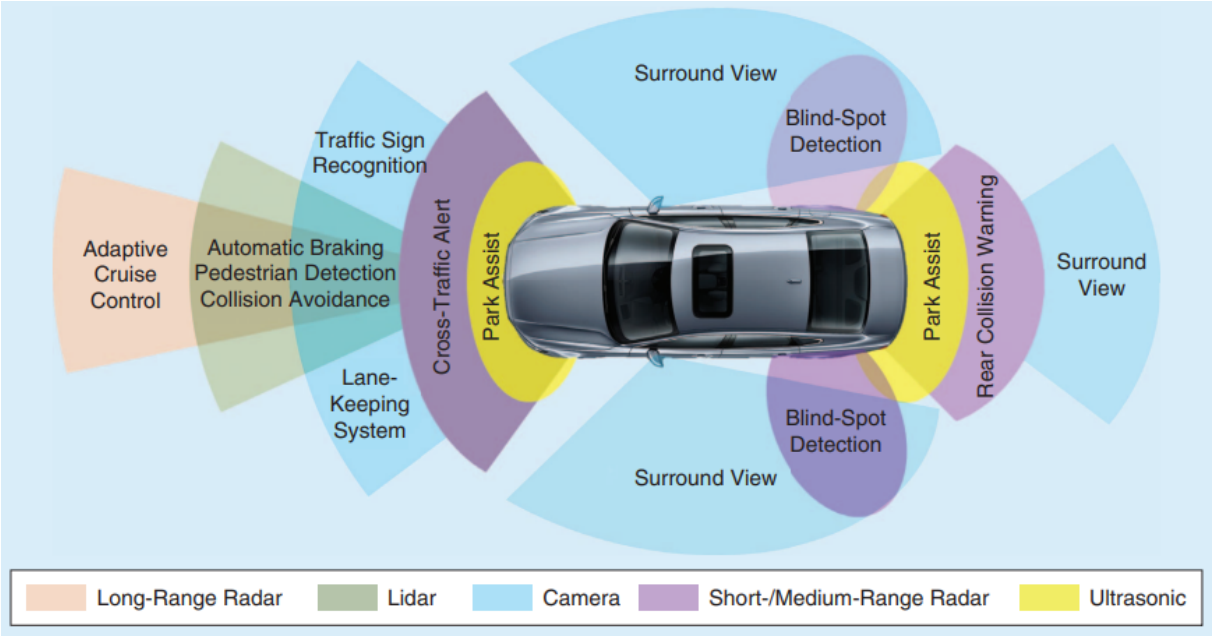

Figure 2.6.: ADAS sensors used in modern vehicles [35]

the vehicle or behind the car for features like the Lane Change Assist (LCA) or Rear Cross Traffic Alert (RCTA) which is designed to avoid accidents while reversing.[37]

Long-range radar (LRR) is measures distances up to 150m and is used in long range collision mitigation features where the driver is made aware of a possible collision or active breaking by the vehicle is engaged if the driver does not respond.

Light Detection and Ranging (LIDAR)

LIDAR uses infrared technology and is used to detect and avoid obstacles and localize the environment. It is based on the technology, that the time it takes the light to travel to and from the vehicle can determine the distance at which the object is located. [38]

Cameras & night vision

Cameras are the most commonly used vision sensor and are used in a mono or stereo version and is able to capture object images and is therefore used in multiple assistance systems like lane detection, traffic sign recognition, obstacle detection or headlight detection. High-end ADAS systems use cameras both outside and inside the vehicle. The technology is becoming more common as a camera can capture information such as texture, color and contrast providing an advantage over other sensor types.
The National Highway safety Transportation Agency (NHTSA) has also introduced a law that mandates vehicle manufacturers to use vision based ADAS which took effect in 2018. [39] Whilst this technology works well during operation in well lit environments, vehicles need to operate in a broad range of circumstances, climates and weather conditions. To account for vision conditions like dusk and darkness or dazzling head lights of oncoming vehicle, a night vision system is required to provide the necessary image quality to obtain quality information of the surrounding environment of the vehicle for ADAS. [40]

Ultrasonic sensors

The distance to an object is measured by utilizing sound waves and are mainly used to detect objects fairly close to the vehicle. Example applications include parking applications like automatic parking of parking assist. [35]

Sensor Fusion

The practice of sensor fusion refers to the combination of information from multiple sensors with the aim of finding the best estimation regarding the current state of the environment. It helps different sensors to complement each others limitations (e.g. in changing environmental/weather conditions) and offers great leverage to the system when compared to the usage of individual sensors. This technology offers

reliability, robustness and high precision compared to individual sensor data and extends temporal and spatial coverage and improved resolution which are crucial in safety critical systems like vehicles. Sensor Fusion does require additional effort and comes at a higher cost and resource consumption but increasing computation power and reduced sensor costs are facilitating the integration of such systems. Different levels and techniques of data fusion and their applications are discussed in [41].

Challenges with ADAS & Emerging technologies

A major problem with current ADAS is the performance of the entire system during changing environments. The system is significantly impacted by different environments and weather conditions. Vision based ADAS for example are severely impacted during foggy or rainy weather conditions and extreme lightning conditions during the night or during thunderstorms. A possible solution for this problem includes sensor fusion which requires obtaining enough input data from multiple sensors and relying on the best possible sensor data from the sensor type that works the best under the current conditions. This would increase the reliability of such systems but also increase costs, making such features likely to be included in high-end vehicles only. [35]
Resource constraints in the automotive sector also greatly impact ADAS. The computation involved in these features involves complex algorithms that result in high thermal dissipation and power consumption. Due to the limited availability of energy in vehicles it is essential for all components to minimize power consumption. This can be achieved by using customized embedded devices and hardware which is specifically designed for lower energy-consumption than traditional central processing units. Furthermore, embedded devices need to operate in real-time as strict timing constraints exist for such devices that control the vehicle. For such use cases optimized hardware and software for minimal power consumption, great performance and reliability are required.

2.2.2. Connectivity and communication

Intelligent transportation systems (ITS) play an important part in today's automotive landscape aiming to making the lives of citizens easier through improved mobility solutions. These systems aim to achieve higher traffic efficiency, providing road and traffic safety, reducing traffic congestion and providing informational and entertainment services. To achieve its goal, such a system needs a way for vehicles to be connected with traditional IT systems. Sensors of the vehicle reach their limitations when it comes to services which require information and data outside of the vehicle. [42] To meet the demand for increased traffic safety and improved traffic flow, data needs to be detected by sensors and shared with other road participants and especially vehicles in vicinity. To enable efficient communication and the

sharing of time critical information between vehicles vehicular ad hoc networks (VANETs) have been developed.

VANETs

VANETs are ad hoc network infrastructures in which vehicles are connected through wireless technologies to communicate with each other. The main responsibility of these systems is to provide effective communication between the vehicle and its surroundings. Studies like [43] demonstrate, that VANETs can be used to enhance traffic safety, reduce traffic congestion and improve traffic flow and driver guidance. They are classified into the categories vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication. VANETs consist of three major components more precisely roadside units (RSUs), onboard units (OBUs) and a trusted authority (TA).
Generally, wireless technology called wireless access in vehicular environment (WAVE) is used for the communication between vehicles and roadside units. WAVE is specified in the IEEE 1609 family of standards describing its architecture, mechanism, set of protocols and interfaces. [44] It helps to increase safety of passengers by providing information on current vehicle status and traffic flow to other participants. Usually, road side units host applications which are used to communicate with other network devices whereas the onboard units mounted on the vehicle gathers useful information like speed or obstacle data. This data can then be forwarded to other participants which are connected to the network of the trusted authority.

Roadside Units (RSU) are computing devices that are fixed in a specific location alongside the road, intersections or a parking area and are used to provide local connectivity for vehicles in close proximity. Due to their fixed installation, they can be fitted with multiple network devices to provide wired connectivity to the internet or other RSUs whilst also providing wireless connectivity through dedicated short-range communication (DSRC).

Onboard Units (OBU) consist of many electronic compionents and are installed in vehicles. A computation device, user interface, sensors and storage are the main components. Their main functionality is to connect to other OBUs or RSUs via wireless technology to obtain and share information about the status of the vehicle and its surroundings.

Trusted Authority (TA) is responsible to manage the VANET system by registering fixed and mobile components and participants. It is a critical component when it comes to the security management of the system as it is also responsible to verify the authentication of participants and needs high computation power to register malicious or suspicious behaviour.

Safety applications based on VANET aim to improve traffic safety by providing safety related information to receivers in time to avoid dangerous situations. Safety related applications are depicted in table 2.4 [45]

Application	**Description**
Traffic signal violation	Alerts about dangerous situations at traffic signals
Intersection collision warning	Warning about intersection traffic
Turn assistance	Assistance in turning the vehicle
Blind spot warning	Alert about another vehicle in the blind spot
Pedestrian crossing information	Alert about pedestrian crossing the road
Lane change warning	Alert about a safe lane change
Forward collision warning	Alert about obstacles
Do not pass warning	Warning that safe overtaking is not possible
Crash alert	Alerts of a crash or nearby accident
Emergency service vehicle	Alerts for clearing a path for emergency vehicles
Curve speed	Warning about sharp curves ahead
Wrong way	Alerts of vehicles going in the wrong direction
Construction site	Warn of construction site ahead

Table 2.4.: VANET safety related applications

By utilizing different networking technologies, VANETs can be used to enable vehicle-to-everything (V2X) communications which play an important role in the field of connected vehicles. V2X communication can be categorized into different mechanisms of information exchange and consists of vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I) and vehicle-to-pedestrian (V2P).

V2V communication requires a transmission mechanism that can provide high transmission rates and short latency so that a vehicle can broadcast useful information about itself and its environment to other vehicles in close to real-time data transmission. [42]

V2I refers to the communication of vehicles with the surrounding infrastructure and is used to transmit relevant information between vehicles and network infrastructures. Since this network infrastructure

does not share the same challenges regarding mobility and resource restrictions as vehicles, the need for larger bandwidth and high computation power can more easily be resolved in such systems.

V2P communication relies on direct communication between vehicles and other road participants like pedestrians, cyclists or other motorized two-wheelers. This communication can either happen directly using VANETs or indirectly through the network infrastructure. It can be categorized into three phases: detection, tracking and prediction and action. Most of these technologies rely on communication with smartphones of pedestrians due to their versatility and the fact, that they are coming with various useful sensors such as GPS, communication and cellular technologies. [46] Nevertheless, also other technologies and communication mediums are currently being researched to allow for enhanced traffic safety in use cases without pedestrians already carrying electronic devices with them.

Radio frequency identification (RFID)

RFID is a technology which identifies objects automatically and collects information by utilizing a reader which uses electromagnetic signals from an adhesive device also known as transceiver microchip. For different applications, different frequencies are used for RFID ranging from a few centimeters to aroung 100 meters. Its application has gained great interest in the automotive industry and can be used for factory automation and integration, access control, supply chain tracking, highway tolls or vehicle tracking. [47]

Bluetooth and Wi-Fi

Bluetooth allows the establishment of short-range connections from one device to another using universal radio interface technology. Applications of Bluetooth in the automotive sector reach from an inter vehicle entertainment facility like pairing the smartphone to the on-board computer to parameter monitoring and vehicle relative positioning.

Wireless Fidelity (Wi-Fi) becomes significantly more used in the automotive industry for monitoring vehicle health, manufacturing and traffic management as the data transmission rate of Wi-Fi is much higher than with Bluetooth or ZigBee. [48]

Ultra-wideband and ZigBee

Ultra-wideband (UWB) is a radio technology with transfer rates around 480 Mbps and low energy consumption with promisiong performance to support intra-vehicular communication systems based on wireless sensors. [49]

ZigBee technology is based on the IEEE 802.15.4 standard consuming low power and minimal expense and despite some limitations, it is considered a feasible option to implement wireless intra-vehicle sensor

networks in small data rate applications. Such networks can be defined as networks of multiple sensors and nodes that work cooperatively in their computation, sensoric and communication abilities.

LTE

Long term evolution (LTE) enables mobile access with low latency and high throughput for a high amount of data and supports broadcast and multicast media deployment. Traditional LTE however faces difficulties when it comes to V2V communication due to its centralized architecture. Due to these observations, LTE-V was proposed by Chen et al. [50] as a systematic and integrated solution for V2X applications. It provides two communication modes complementing each other: LTE-V-Direct is designed with a decentralized architecture to provide direct V2V communication with low latency and high reliability whilst LTE-V-Cell aims to support the V2I use case in utilizing a centralized star topology.
To make ITS a reality, it is essential to provide reliable, efficient and in-time communication abilities to all vehicles, devices and their sensors. Fifth generation mobile networks (5G) provide possible solutions to many challenges that ITS face today by utilizing advanced technological features like higher capacity and data rate, massive device connectivity and consistent quality of service. [51]

Cloud based technologies

The development of connected vehicles to enhance traffic management, passenger safety and improved infotainment systems is facing various challenges like scalability, connectivity and flexibility issues as well as resource constraints in vehicles themselves. To overcome these limitations, cloud computing technologies are becoming of more and more importance by analyzing and storing data and providing effective decisions for connected vehicles. [52] Conventional cloud computing technologies are not suited in a high mobility and low latency environment so technologies like edge cloud computing are a viable option for such use cases. [53] Edge cloud computing allows optimal resource utilization and better latency by performing computations on available nearby RSUs, base stations or other infrastructure.

Over the air (OTA) updates

As vehicles are becoming more connected and have increased functionality like ADAS or self-driving modes, more hardware and software is required to enable this functionality. These new functions of connected vehicles and the corresponding services are deeply related to the software developed in association with the hardware of these vehicles. This in turn has also increased the number of software parts and increased the lines of code within a vehicle significantly. Additionally, software malfunctions can have serious consequences and can involve accidents which threaten human lives. [54] In traditional solutions,

software updates are conducted when the driver comes to a garage or dealership for a service or in more critical cases, vehicles were recalled by the manufacturer. This process is not only time consuming as during the time until a fix can be provided, the vehicle owners would need to operate their vehicles with bugs or flaws, but also very cost intensive as an increase in the software portion in the vehicle would also increase the amounts of faulty code that might be discovered after Start-of-Production (SOP). Furthermore, new concepts like paying for specific vehicle functionalities per feature like Software-as-a-Service (SaaS) are transforming the automotive industry into a more software-driven mobility service defined by the experience whilst transporting passengers. [55]

The software industry is using over-the-air OTA updates for software upgrades on computers, mobile devices and even televisions for many years but OTA which corresponds to a wireless delivery of software updates to vehicles is still rare in the automotive industry. OTA update is a method of remotely providing and distributing performance and maintenance updates and feature enhancements throughout the lifespan of a vehicle. OEMs are able to manage the upgrade process for individual ECUs via wireless connectivity solutions. The benefits of this technology are huge and would solve many problems. It eliminates the need for a garage or dealership visit, saves the time which passes whilst waiting for updates, protects brand image and would save billions of dollars according to studies if vehicles no longer need to be recalled. [56].

Some vehicle manufacturers are already using OTA update systems, but have focused on non-essential components like infotainment systems, maps and navigation or telematics software. Tesla has already ventured into software updates of individual ECUs with safety-critical features like the breaks. This process is known as Firmware Over The Air (FOTA). ECUs within the vehicle operate on semi-permanently stored software (Firmware) on its hardware that controls the system. Firmware is commonly associated with embedded devices as the used hardware requires tailored software.

The development of an OTA mechanism for all components seems like an obvious decision, but security concerns and potential vulnerabilities need to be solved before an industry wide adoption can take place as history has shown, that software systems are vulnerable in safety and security aspects when undergoing change. [57] Some manufacturers already encountered problems during the process. An OTA update by Lexus in 2016 for example was responsible for an unresponsive infotainment system. [58]

Remote diagnostics

Modern vehicles contain a number of electronic subsystems such as fuel injection, gear box, breaks, etc. which are managed by ECUs which are connected to numerous sensors and actuators to control vehicle functionalities. The entire system relies on these electronics and safety applications and strict regulations

are in place which require accurate monitoring and repair. [59] Failure of a single component can impact the entire powertrain as for example missing data points within the exhaust system can force the engine into a limp-home state. As vehicles are becoming more complex, there is critical need for the proper detection and reporting of failures within the vehicle.

A unified solution to this problem is standardized by ISO 14229 [60], Unified diagnostic services (UDS), specifying data link independent requirements of diagnostic services, which allow a diagnostic tester to control diagnostic functionalities. [61] It has no restrictions on the utilized physical communication medium and can be used for applications like calibration, diagnostics and reprogramming. UDS specifies the diagnostic services which can be used to diagnose various functionalities. This enables technicians to test an read out relevant data from individual ECUs by communicating with them over the CAN bus or other communication mediums and requires that applications are present on all the relevant ECUs so that communication with the UDS tester is possible. As an illustration, if the correct operation conditions are met, an UDS service called 'ECU Reset' can reset the addressed ECU if need be.

UDS is used to monitor parameters of different vehicle components, writing data into non-volatile memory or testing various features or Input/Output (I/O) pins enabling technicians to diagnose various vehicle functionalities and can be divided into six functional units:

Diagnostic and communication management defines the main session control service and communication timing parameter settings which allow services to run into supported sessions.

Data transmission monitors the parameters of a vehicle with an unique Data Identifier for each parameter.

Data storage is used by diagnostic tools clear Diagnostic Trouble Codes (DTCs) or other diagnostic information from devices memory.

Input/output control is responsible for controlling the I/O pins of an ECU. This can be used for testing as it provides direct control over the I/O pins of the device. Performance of control devices such as actuators, fans and control algorithms which depend on input signals can directly be checked.

Routine Control is used to perform project or OEM specific functionalities with individual identifiers which can be used for starting or stopping the execution of any task within the ECU such as erasing non-volatile memory or performing checksum calculations, etc.

Upload/download enables writing or reading of firmware from the memory of hardware devices and can be used to program or verify the firmware on an ECU.

Current implementations rely on a device also called Tester to be directly connected to the vehicle generally using the CAN protocol. The introduction of connected vehicles with internet access now also allows manufacturers to enable diagnostic services remotely. BMW as an example has introduced a remote diagnosis feature in cars fitted with their operating system iDrive 7 OS. [62] By utilizing cloud technology, BMW is able to diagnose their vehicles from a distance in case of an error. If an internet connection is available, a BMW representative can remotely connect to the vehicle and check its status, faults and advise on further actions without the need to study the manual and consecutively drive to a repair shop or service center to connect a diagnosis tool to provide information about the exact failure. This enables service centers to pre-order repair parts and time at the dealer can be reduced.
Research in this field also demonstrates the possibility for more advanced features like Shafi [63], who proposes a method to predict vehicle maintenance based on machine learning algorithms enabled by today's technologies. The proposal collects live sensor data from the vehicle in normal condition and relies on data obtained by vehicles in a faulty condition. The data is transmitted to back end servers who analyze the data based on machine learning technology. Furthermore, these patterns can be analyzed and compared to live sensor data in order to find similar behavioural characteristics to predict failures and optimize maintenance.

Internet of Things (IoT)

IoT is considered as a blessing for many industries, including automotive to offer opportunities in developing, creating and enhancing services for an increasing user base.[48] It has significantly expanded in the automotive industry enabled by increased connectivity and applications. Embedded systems and networking infrastructure are interconnected and provide end users the ability to accumulate, access, transfer and process data over the internet. The technology can therefore play a significant role in improving overall transport efficiency and driving comfort. It also enables productivity improvement within manufacturing, real-time performance monitoring and vehicle diagnosis and maintenance.

Real-time vehicle navigation can have a significant impact on ensuring smart and safe transportation by providing the driver with live traffic information, public transport tracking or even notifying the nearest rescue team for an emergency response in case of an accident. [64]

Traffic management is one of the most important aspects of mobility solutions today and IoT can enable smart traffic management systems to improve road safety. Vehicle density can be used to intelligently time traffic signals, information on accidents can reroute traffic to avoid congestion and the time spent searching for parking spots can significantly be reduced which is not only

beneficial for drivers, but also helps the environment. Even accidents due to vehicle height can be avoided by deducing physical characteristics of the vehicle before height restrictions lead to issues. Payment for toll charges for example can also be improved by the utilization of IoT.

Security / Anti-Theft IoT is able to improve security features of vehicles by providing the ability to track a stolen vehicle and determine its location via the internet or even stop the car remotely. A paper by Shreyas et al. [65] shows a mechanism to capture an image of the thief as digital evidence.

Event Data Recorder can provide features such as providing information on accidents or unexpected incidents when they occur and help in solving insurance and criminal cases. Germany has introduced a law requiring vehicles with a specific automation level to implement an event data recorded. [66]

Emergency vehicle management can improve response time of emergency services significantly by first of all notifying the nearest possible responder and consecutively utilizing traffic data to find an optimal route to the desired destination. Vehicles on that route could also be notified about the situation, warned about emergency vehicles approaching and instructed to make way for emergency services.

Pollution monitoring is enabled through IoT by evaluating the vehicular exhaust pollutants at specific locations or inform vehicle owners about their emission-quality and footprint.

Driver physical condition monitoring can be enhanced by IoT by improved ways to observe the drivers physical condition like customizable observation sensors or connecting additional apps or appliances like smart watches to the surveillance system.

Vehicle performance monitoring can evaluate data from vehicle sensors and detect irregularities or the need for maintenance during operation. By utilizing cloud computing in such a system, maintenance of certain components can even be predicted before the system runs into issues.

2.2.3. Autonomous/ Cooperative Driving

Gartner defines "an autonomous vehicle is one that can drive itself from a starting point to a predetermined destination in “autopilot” mode using various in-vehicle technologies and sensors, including adaptive cruise control, active steering (steer by wire), anti-lock braking systems (brake by wire), GPS navigation technology, lasers and radar." [67] However, according to leading experts at BMW [68] autonomous would better be described as "self-sufficient" in a sense that the vehicle will operate independently of human intervention and a driver does not necessarily need to constantly control or monitor

the system. This definition is better suited due to the fact, that vehicle is not "self-governing" when in self-driving mode. It is directed by code that is written by engineers to pre-determine the response of the vehicle to the environment.

What seemed like science fiction for a long time is becoming a reality in today's automotive landscape. Cars with self-driving mode are on the forefront of mobility research and constantly in a spot light in the media. Already today, cars offer various features like steering assist or lane assistance and technologies and ADAS to prevent collisions have become reality. Prototypes of partly and fully automated cars are in the test phase on public roads. [69]

Numerous high-tech sensors detect the current position of the vehicle and its environment and navigation. Artificial intelligence in the form of software enables the vehicle to analyze and interpret the data collected from these sensors based on complex algorithms. The development and testing of functionalities makes the system increasingly smarter as the software learns from recognizing objects and situations and is able to make better and more accurate decisions. Automated Vehicles can be categorized into different levels of automation. By understanding the differentiation between the different stages it is possible to understand the technology which paves the way to fully automated driving better. The Society of Automotive Engineers (SAE) has classified six automation levels for road vehicles (SAE level 0-5). [70] During the lower-levels (0-2) the driver monitors the driving environment and steers the vehicle. In contrast, in the higher-levels (3-5) the environment is monitored by sensors and ADAS systems and steering is done by the vehicle in certain traffic conditions. As a reference, the 2016 Tesla model S with top-of-the-line ADAS is level three and multiple safety relevant components are handled by the vehicle and the driver is required to intervene when needed.

0. No automation supports the driver, only informative features are active.

1. Driver Assistance Systems support the driver, e.g. in regulating the distance to other road users. The driver still has to execute all main operations like braking and steering.

2. Partly Automated Driving where the car system partly executes operations like braking, steering or acceleration (e.g. steering and lane control assistant). The driver still has the full responsibility over the vehicle and has to monitor the automated function as well as the environment at any time.

3. Highly Automated Driving implies that the vehicle is able to drive autonomously in defined traffic situations, for instance on highways. The driver could do other things, but must still be able to take over control if necessary.

4. Fully Automated Driving is possible and the vehicle can even handle highly complex traffic situations without any human intervention. Anyway, the car still has a cockpit and drivers have the

possibility to interfere. Even though the driver could temporarily execute other tasks like sleeping or reading a book, his/ her presence is still required.

5. Full automation: A car can drive fully autonomously and can perform any action without human interaction or attention. At this level of autonomy, humans are only passengers and do not even need to have a driver's license.

As autonomous vehicles (with full automation) respective of SAE level 5 would be considered vehicles that are full-time operated by an automated driving system without any interference from a human driver and are therefore not expected for at least the next 20 years.[68] Therefore this paper will further consider vehicles with self-driving mode.
To provide a high-level overview of the functionalities and information that needs to be available for a self-driving mode to be possible, vehicles can no longer rely solely on information provided by its sensors. The following list provides an overview of required technologies to enable self-driving mode.

Driving order Human interaction is required to tell the vehicle its destination. The input mechanism can be the on-board navigation system or smartphones and apps.

GPS Most transport positioning systems today use the Global Positioning System which can guarantee an accuracy of positioning with deviations below eight meters, which is not sufficient for self-driving technology

DGPS With the usage of reference stations around the globe, GPS inaccuracies can be eliminated

Landmark/marker orientation Using vehicle sensors to recognize prominent landmarks and road markings like the center line can be used to further improve the positioning and furthermore aid orientation.

Environmental sensors By using chips or electronic mechanisms stored in the road or its environment with corresponding transmitter-receiver technology the vehicle can electromagnetically detect its self-position.

Vehicle data By using vehicle speed and direction of travel, the vehicle can more accurately calculate its position

Vehicle status sensors Different sensors provide information about the current status of the vehicle including system status, tire pressure, thermometer, velocity, etc.

ADAS sensors Most importantly, the vehicle needs information about the environment, roadway, traffic and weather conditions, surrounding infrastructure and other road participants. All sensors covered in section 2.2.1 provide information about surrounding objects.

Communication services Additionally, to satisfy the high amount of data required, the vehicle must also communicate with its environment and the internet. Wi-Fi can be used in close proximity to infrastructural components or traffic participants whilst mobile communications like the 5G mobile standard seems suitable for an internet connection with high data and low latency transmission. Local digital map and traffic data and weather reports can be downloaded from the internet providing even more accurate information of the current situation. Vehicle-to-X communication can also be used to supplement additional data (e.g. hazards around corners otherwise not visible to the sensors)

Data consolidation and evaluation All this acquired data needs to be collected and processed in a high performing processing unit to calculate the best coherent picture of the situation.

2.3. Securing Electronic Control Units

Automobiles have transitioned from a mere mechanical system to a complex system consisting of various computers and embedded devices as control systems. These systems coordinate driving and comfort functionalities across independent ECUs by communicating over one or more vehicle internal bus networks. Whilst safety aspects were always a critical engineering concern and a focus topic of development activities, the design by vehicle manufacturers did not take security considerations of the increased attack potential into account. [71] New standards and regulations, such as a mandated On-Board Diagnostics (OBD-II) port in basically all modern vehicles which provides direct access to the internal vehicle network also raises security concerns. Additionally, modern vehicles are becoming increasingly connected with interfaces to lots of different systems including VANETs, near-field communication, LTE and V2X. This connectivity enables vehicles to send and receive a variety of information which in turn also has severe impact on its security as new vulnerabilities are introduced. These trends suggest that a broad range of attack vectors will be available to attackers which can compromise vehicle components or the vehicle network with severe consequences.
This chapter aims to provide an overview of current security challenges in the automotive field by analyzing current developments and emerging technologies.

2.3.1. Automotive security and challenges

This section discusses key challenges which are specific to IT application in the automotive industry and difficult to manage in comparison to other industries. We have already established, that connecting the car to the external world poses new risks and challenges from a security perspective, but it needs to be considered that in the automotive industry, a security issue can directly relate to a severe safety issue with the potential to harm human lives. This section discusses key security challenges that apply to the entire automotive industry since ECUs were introduced to vehicles to provide a better understanding of the circumstances in the automotive landscape and differences to traditional IT infrastructure.

Massive Scale and Density

As vehicles are becoming more connected and digitized, they are often compared to traditional IT systems, referring to them as "computers on wheels". However, compared to computers vehicles consist of numerous individual ECUs and resemble the scale of an IT infrastructure of a small enterprise network with all the embedded systems, devices and sensors packed into a mobile chassis. Therefore, managing cyber security aspects of millions of mobile vehicles is a scale not common in a traditional IT environment. [72]

Regulations Requiring Cyber Security Type Approval

Manufacturers in the IT industry are usually not directly responsible for the cyber security of their products. The users and enterprises have to implement the right tools and measures to protect their network and data. Data privacy laws like the General Data Protection Regulation (GDPR) mandate security measures from enterprises and service providers to protect the customer data that is processed.
However in the automotive industry, cybersecurity needs to be implemented furing the manufacturing stage and OEMs are directly held accountable for flaws in their cyber security implementation. The first regulation to require cyber security measures during the entire supply chain and vehicle lifecycle is UNECE WP.29 [73] which means that vehicles need to adhere to a type approval process.

System complexity

Besides the already mentioned scaling issues, vehicles consist of thousands of hardware components coordinated by numerous ECUs operating on millions of lines of code. Additionally, all the devices serve a unique purpose and are not running on a universal operating system and the failure of one of them can lead to the entire vehicle not starting anymore.
Furthermore, ECUs are highly sophisticated embedded devices and are often manufactured by many

different suppliers that specialize on specific functionalities which in turn often also rely on third-party suppliers to assemble their product. This implies, that OEMs need to manage cyber security across different organizations and align all development activities so all components are interoperable.

Long Lifespan

With increasingly efficient engines and mechanical systems, vehicles are lasting longer than ever and have an approximate lifespan of around 10-20 years according to studies. [74] This corresponds to a lifespan three times larger than consumer electronics according to estimates. [**?**] Long-lasting cars pose additional challenges to manufacturers as managing software updates becomes increasingly difficult, especially when considering the mobility aspect of automobiles compared to stationary traditional IT systems.

Scattered Locations

The mobility aspect of vehicles also becomes a cyber security challenge as incident response activities face unique challenges compared to traditional IT systems. Furthermore, geographical and environmental challenges result in connectivity issues which complicates the management of these systems.

Damage Severity and Recovery

In case of a successful cyber attack, vehicular systems not only put data at risk, but also the personal safety of the driver, passengers and other traffic participants. Furthermore, in case of an emergency, vehicle owners can not switch to a backup vehicle like it would be possible in many cases of traditional IT infrastructure.

Resource restrictions

ECUs are embedded devices and therefore often face challenges when it comes to resource allocation and utilization. [75] Software in embedded devices needs to be designed with resource restrictions in mind adding not only complexity, but also financial aspects in the decision making process which controller to use. Traditional cryptographic libraries used by most IT systems need to be optimized for their use in embedded devices.

Real-time requirements

Many ECUs have hard real-time requirements and resource intensive cryptographic operations can not interrupt engine or breaking functionalities adding additional complexity in designing security solutions

for such systems.

2.3.2. Security in Depth

As single countermeasure is considered as insufficient when it comes to cyber security as even under the assumption that a conceptual measure is without flaws, errors or software bugs can still be present and exploited by attackers. Considering that modern vehicles consist of million lines of code, the remaining attack potential becomes evident as bugs remain one of the main drivers for vulnerabilities. Therefore, a layered security approach is required so the exploit of a single vulnerability does not lead to a compromisation of the entire system. In traditional IT systems, this layered approach consists of firewalls, network zones and separation, etc. and is commonly used to minimize the impact of a cyber attack.
Figure 2.7 shows a multi layered architecture by Schnieder [76] consisting of five different security areas.

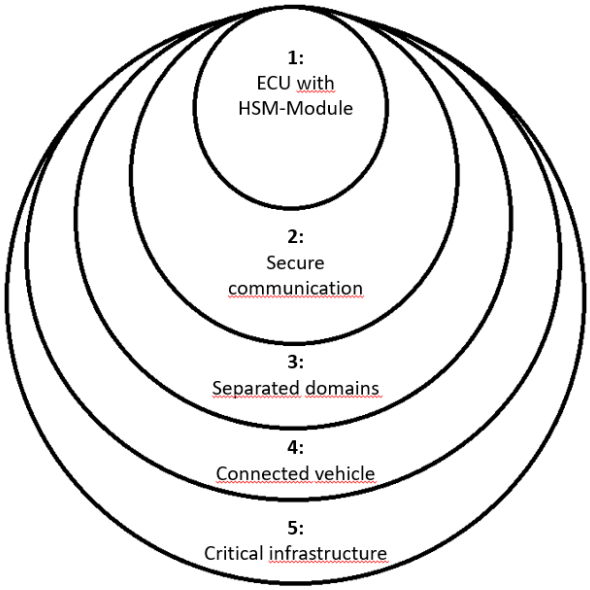

Figure 2.7.: Layered security approach [76]

Layer 1: Securing individual ECUs is the foundation for a successful security architecture. Debug ports, diagnosis and flash/update procedures need to be protected against unauthorized access to protect the integrity of these devices.

Layer 2: The security of vehicle internal communication for important messages needs to be considered to prevent the manipulation or forging of such messages. Security mechanisms include the authenticity of sender and receiver via cryptographic keys, the integrity via cryptographic algorithms and validity of messages via timestamp or counter mechanism.

Layer 3: Secured domains within the vehicle architecture are required to separate important components and reduce the impact of compromised devices. The utilization of dedicated gateways secures the architecture by authorizing and authenticating access to central bus systems. Intrusion detection systems can aid in the detection of malicious activities, although automated reactions to such incidents need to consider safety requirements.

Layer 4: The security of functionalities and interfaces in connected vehicles communicating with external components increases the attack surface immensely. Firewalls can help to mitigate risks of unauthorized access from outside the vehicle.

Layer 5: Security of critical infrastructure components becomes of great importance when considering V2X communication. Navigation and lateral/longitudinal control mechanisms may depend on data received from network infrastructure in self-driving vehicles. Unauthorized access to traffic signals for example can impact traffic safety and needs strong security measures.

2.4. Related Work

This section provides an overview of relevant work which is related to this paper or referenced in this thesis.

2.4.1. Threat and Risk Analysis (TARA)

Traditional security risk assessment methods for IT-infrastructure are not taking safety implications into account, which are a serious deviation when it comes to the potential impact of cyber attacks on automotive systems. [77] A review of the available TARA methods in the automotive industry was conducted by Macher et al. [78] which consists of an analysis of TARA methods for early development phases in context of ISO 26262 (Functional Safety Standard for Modern Road Vehicles) and SAE J3061 (Cybersecurity Guidebook for Cyber-Physical Vehicle Systems). The researchers concluded, that current security standards and guidelines were still incomplete and not directly applicable in practice. However, at the time of the TARA method review, the recently published standard ISO/SAE 21434 has not been released, which also references frameworks for threat modelling which are suitable for the threat scenario identification according to the standard.

EVITA

This TARA method is part of a research project funded by the European Commision (EVITA - E-Safety Vehicle Intrusion Protected Applications) and considers threats for a particular feature from a functional

perspective.

TVRA

Is considered as not applicable for cyber physical systems in vehicles according to Macher [78]. It aims to identify assets in the system and models the likelihood and impact of attacks. However, the method was developed for telecommunication networks and does not seem suitable in the automotive domain.

Process for Attack Simulation and Threat Analysis (PASTA)

The Process of Attack Simulation and Threat Analysis (PASTA) is a risk-centric threat modeling methodology aiming to provide a process for simulating attacks to applications and analyze/mitigate cyber threats. [79] It is a seven stage process but focuses on the evaluation of applications and is not designed for cyber physical systems.

STRIDE [80]

Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege would be a suitable mechanism for the analysis of this paper and is used in the HEAVENS [81] and SAHARA [82] methodology discussed in the review by Macher et al. [78] but in itself does not consider safety relevant aspects.
HEAVENS seems to be well suited to fit our use case but represents a wider scope and can be considered as an entire security model rather than a threat and risk analysis method.

Attack trees

Attack trees display different ways how a system can be compromised with the focus on an attacker's point of view. They offer a methodical and systematic way to show the security of a system, based on different attack scenarios. [83] Attack trees are a structured and graphical representation of threats against a system in form of a tree notation. The attack goal is represented at the top of the tree structure as the root node. There may be multiple root nodes representing different goals. A root node then gets decomposed and further compartmentalized into subtasks. The idea behind this principle is, that understanding various threats can be achieved by simplifying attack steps through subdivision. The attack tree methodology is ordered in a hierarchical fashion, where nodes of nodes are referred to as child nodes and the nodes above them are called parent nodes. The branches of the tree represent the attack paths, leading to the attack goal or threat. When connecting the nodes, each connection can be represented by OR/AND. AND means that every connected sub-goal of that node needs to be satisfied,

whereas with OR, a single sub-goal is enough to achieve the goal. For example, an authentication with username and password needs both values to be correct in order to log in successfully, but when resetting the password of a user, it's sufficient to either provide the username or the email address. An example illustration of an attack tree including multiple attack paths leading to the overall goal to open a safe is illustrated in figure 2.8.

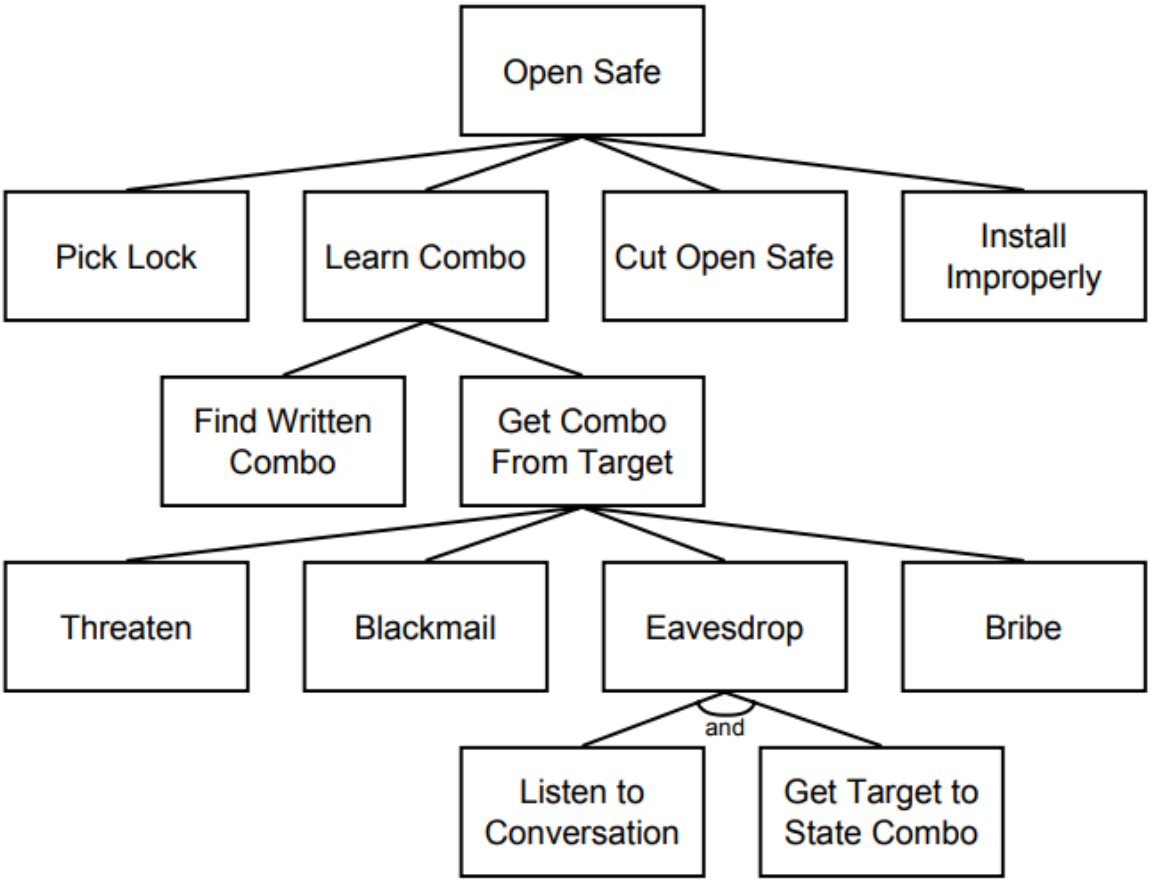

Figure 2.8.: Example attack tree [83]

2.4.2. ISO/SAE 21434

The automotive industry has recognized the lack of a cyber security standard and is aware of the new challenges regarding automotive cyber security which have emerged and therefore invested in the development of an industry standard to protect assets and tackle automotive cyber security issues. [84] The joint working group of SAE and ISO have published an international standard "ISO/SAE 21434: Road vehicles - Cybersecurity engineering" to address the cyber security perspective in the engineering of electrical and electronic (E/E) systems within road vehicles. [13] By ensuring appropriate consideration

of cyber security, the document aims to enable the development of E/E systems to keep up with evolving attack methods and state-of-the-art technology.

The standard neither specifies cyber security technologies, remediation methods or solutions and does not include unique requirements for self-driving vehicles or road infrastructure. Instead, a risk-oriented approach for the methodical elicitation and prioritization of actions for cyber security measures is encouraged. [84]

ISO/SAE 21434 Overview

The key focus of ISO/SAE 21434 are cyber security activities regarding all vehicle life-cycle phases beginning with design and development to production, maintenance and operation and finally decommissioning. A detailed analysis of the structure and the different sections covered by the standard are illustrated in figure 2.9 and a brief description about the activities in each section is provided.

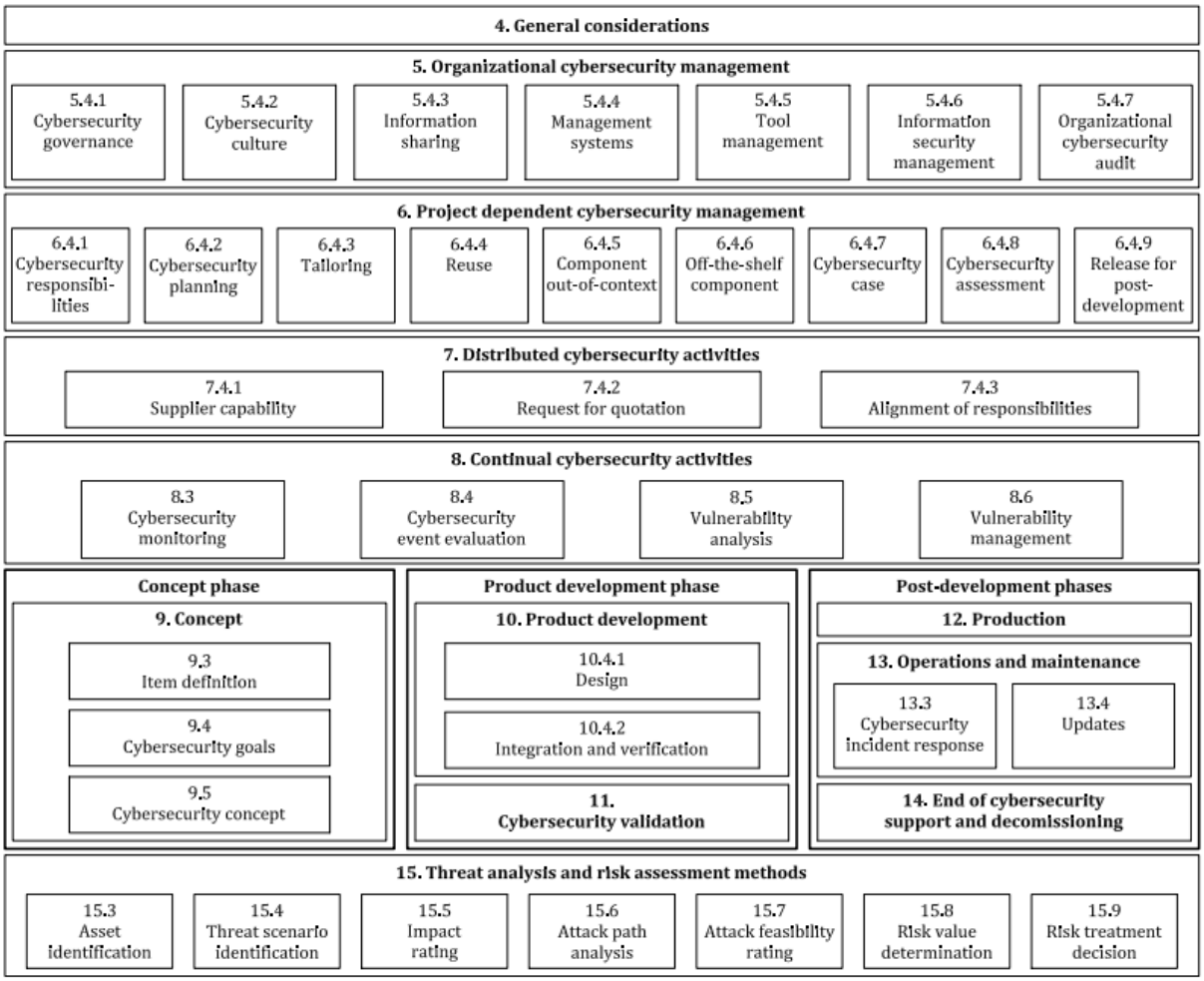

Figure 2.9.: ISO/SAE 21434 structure overview [13]

Section 1 defines the scope of the document, described in the subsection 2.4.2.

Section 2 includes normative references.

Section 3 defines terms, definitions and abbreviated terms which are used throughout the document.

Section 4 is informational, describing organizational cyber security management, the vehcile ecosystem and context and approach taken in the document.

Section 5 specifies the organizational cyber security strategy including management, policies, processes and rules.

Section 6 describes cyber security management and activities at project level

Section 7 outlines requirements for cyber security activities and responsibilities between customer and supplier

Section 8 defines continuous cybersecurity activities that provide information on risks and vulnerability management

Section 9 includes activities for the determination of cyber security risks, goals and requirements for an item

Section 10 defines product development activities with security relevance like cyber security specifications, implementations and verification

Section 11 describes the cyber security validation of an item at the vehicle level

Section 12 includes cyber security relevant aspects for production, manufacturing and assembly of items or components

Section 13 defines activities for cyber security incident response and updates to components or items

Section 14 includes considerations for the decommissioning and end of support for items and components

Section 15 defines risk management requirements including a methodology and plan to determine circumstances and events which may lead to cyber security risks so that treatment can be pursued

2.4.3. SAE J3061

SAE J3061 [85] or "Cybersecurity Guidebook for Cyber-Physical Vehicle Systems" provides high-level guiding principles and a cyber security process framework and guidance for organizations to identify and assess threats to design cyber security into cyber-physical vehicle systems in the entire vehicle development lifecycle process. It is considered a foundation for further standard development and is superseded by ISO/SAE 21434 2.4.2.

2.4.4. UNECE WP.29

WP.29 is an international automotive cybersecurity regulation to mitigate cyber security risks posed to passenger vehicles. It establishes performance and audit requirements for cyber security and software update management for new passenger vehicles sold in participating countries including nations of the EU, UK, Japan and South Korea. Manufacturers selling in these regions also need to consider the implications of the regulation. The UN regulations are legally enforceble and need to be adhered to in order to achieve type approval to sell into the market. [86]
The regulation concerns the approval of vehicles with regards to cyber security and cyber security management system [87] and the approval of vehicles with regards to software update and software updates management system [88]. The annex within these documents defines threats to vehicle functionalities and defines requirements for mitigation scenarios but does not go into further detail to define specific countermeasures or implementation guidelines.

2.4.5. Auto-ISAC

The Automotive Information Sharing and Analysis Center (Auto-ISAC) is a proactive collaboration to protect traffic participant safety though cyber security aiming to provde best practices for security the vehicle environment and provide implementation guidelines. The executive summary released in 2016 is a high-level document defined key cyber security functionalities and best practices but is not intended to provide specific action or measures. [89]

2.4.6. JasPar

The Japan Automotive Software Platform and Architecture (JasPar) was established to enable standardization of electronic control systems and software of in-vehicle networks, thereby allowing industry-wide common implementation, more efficient development and increased reliability. [90] it aims to identify common issues that the automotive sector will face in the future to resolve them and create common

objectives throughout the industry. Members can download information about requirements and specifications for vehicle technologies and security features from the website. Different working groups are established for different activities including cyber security.

3. Requirements Analysis

This chapter explains the proposed approach of this paper in detail and aims to provide a clear picture about the used methods, requirements and scope. Assumptions and boundaries are discussed (but clarified in further detail in the implementation section 4) in order to provide an understanding of how this paper systematically reaches its goal of answering the research question.

3.1. Research question

Which security measures are required to secure electronic control units against threats from emerging vehicle technologies?

3.2. Overview

This paper aims to determine security requirements and measures for individual ECUs by using a systematic approach. The proposed approach follows a tailored methodology proposed in the ISO/SAE 21434:2021 standard to determine threats and risks to ECUs, elicitate requirements and define countermeasures to reach an acceptable risk level for the identified cyber security threats. To clarify the intended goal of this thesis, the terminology used to form the research question is explained below:

Security measures define the countermeasures to cyber security threats found during the threat and risk analysis, an example of such a security measure would be the introduction of a certificate based authentication scheme to secure the ECU against unauthorized access.

Electronic control units (ECUs) are embedded computers in vehicles to control the electronics of mechanical systems or components (2.1.1). During the item definition phase of the analysis, an exemplary ECU is defined in accordance with the selected methodology defined in the ISO/SAE 21434:2021 standard. However, the identified security measures are representative for all ECUs in the vehicle network. It also needs to be considered, that ECUs with specific applications, like the telematics unit responsible for vehicle external communication will require an additional analysis based on its unique feature set.

Threats are defined as "Any circumstance or event with the potential to cause the security of the system to be compromised." [91]

Emerging Vehicle Technologies are technologies and innovations in the automotive sector with the aim to make vehicles safer, more connected and driver-less. The technologies and features relevant for this paper are determined based on a literature analysis in chapter 2.2.

In order to consider the research question as answered, the threats identified during threat and risk assessment are assessed utilizing an attack path feasibility rating described in more detail in the methodology 3.4. The threats are then evaluated to define countermeasures which reduce the risk to an acceptable level.

3.3. Scope

This section clarifies which areas and components are focused on in this approach and provides reasoning, why certain areas - even though security relevant in the perspective of cyber physical systems - are not covered in detail.

Since modern vehicles are very complex systems, a detailed analysis of connected vehicles including the surrounding infrastructure is infeasible with the available time and resources and is therefore not the focus topic of this paper. In order to determine specific software- and hardware requirements for ECUs, layer 1 in the layered security approach 2.7 will be the focus of this work. A detailed overview of the Target of Evaluation (ToE) is provided in 4.1.

Unique features and ECUs for specific use cases such as the gateway or telematic control unit share the same base requirements regarding security, but unique security measures for such system are not discussed as they are application specific and not relevant for ECUs in general.

Components like sensors, actuators and multimedia systems are not comparable to ECUs and will not be evaluated in this work. The vehicle network is part of the analysis due to its interfaces to the system of interest, but the network itself in particular is also considered out of scope of this thesis. Functional safety aspects are considered during the analysis due to their relevance regarding severity impact for the risk analysis but are not analyzed in further detail as this paper has its focus on security aspects.

Use cases for features identified in the literature analysis in chapter 2 will be evaluated, but security threats will only be considered if they are relevant for the security of individual ECUs. The evaluation, if relevance for individual ECUs exist will be based on the asset identification during the threat and risk analysis.

3.4. Methodology

Innovations and features which were identified during the literature analysis in chapter 2 will be systematically analyzed and cyber security threats will be derived by defining use/misuse cases that these new functionalities introduce. A threat and risk analysis method proposed in the recent standard ISO/SAE 21434:2021 will be used to identify the most prominent risks of such a technology so that concrete security measures can be defined to treat the identified risks. This work focuses on security measures specifically for ECUs to provide detailed technological countermeasures instead of high-level security aspects for the entire vehicle due to the complexity of today's automotive systems.
As already outlined in the scope 3.3 the focus lies on threats to individual ECUs and does not consider the entire framework that ISO/SAE 21434 provides. Nevertheless, a tailored approach utilizing the standard can be used to fulfill a systematic analysis of threats and risks. The proposed approach will consider section 9 of the ISO/SAE standard for the determination of cyber security risks and goals.
This concept phase involves the consideration of vehicle functionalities which are implemented in specific items. Therefore, a clear definition of the item, its boundaries, functionalities, operational environment and preliminary architecture is required to provide the basis and scope for all subsequent activities. Additionally, some assumptions and limitations regarding the item and its environment need to be defined and considered for the analysis phase. Following the item definition, an analysis needs to be performed involving the methods and procedures outlined in section 15 of the standard describing threat and risk assessment methods and procedures. The next step in this approach involves the definition of security goals based on the risk treatment decision. These security goals can furthermore be used to define technical and/or operational security controls to achieve the defined goals. Furthermore, as suggested in SAE J3061 (Cybersecurity Guidebook for Cyber-Physical Vehicle Systems) the identification of countermeasures will be based on the TARA. "The TARA results will drive future analysis activities by focusing future analyses on the highest risk Cybersecurity threats." [85] After the identification of appropriate countermeasures for the determined threats, the remaining risk will be reevaluated in a risk assessment considering the mitigation effectiveness of the implementation of these security measures.
The TARA will follow the following order, as also proposed in the example provided in ISO/SAE 21434 and also corresponds to the document structure in the implementation section:

1. Item definition
2. Asset identification
3. Damage scenario impact rating

4. Threat scenario identification
5. Attack path analysis
6. Attack feasibility assessment
7. Risk evaluation

3.4.1. Threat modelling approach

Different methods of TARA approaches are discussed in the related work section 2.4.1 whilst ISO/SAE 21434:2021 mentions frameworks such as TVRA, EVITA, PASTA and STRIDE. This paper follows the first approach proposed in the standard: An elicitation of malicious use cases based on reasonably foreseeable misuse and/or abuse [13] because it is considered the best method for the analysis conducted in this work due to its suitability for the proposed approach and the limitations of the other frameworks: Whilst the EVITA method is regarded as suitable, its classification for severity is adopted and therefore not conforming to ISO 26262 [92]. TVRA is not considered as suitable for cyber-physical systems in vehicles since it was developed for data and telecommunication networks according to the analysis done by Macher et al. [78]. PASTA was also not intended for use in the automotive environment but rather suits traditional IT applications. [79] Lastly, the standard refers to STRIDE as a suitable threat modelling approach but this method is considered less suitable for the application in this paper as it does not consider safety aspects. [80] Additional information regarding the application of the TARA is provided in more detail in the implementation chapter 4.

3.4.2. Risk mitigation

The outcome of the threat and risk assessment are threats and their corresponding attack feasibility rating. These ratings will be considered during risk evaluation to define how each individual threat shall be treated. The risk treatment decision will be done in accordance with the method proposed in ISO/SAE 21434.
If the treatment decision results in the need of a mitigation measure, appropriate countermeasures for these risks will be identified and described.

3.4.3. Countermeasure evaluation

A security measure will be considered as sufficient, if the underlying risk can be mitigated so that the remaining risk can be accepted. In order to evaluate the effectiveness of the mitigation measure, the

risk evaluation will be repeated under the consideration of the countermeasure implementation until an appropriate risk level is achieved.

4. Implementation

4.1. Item definition [RQ-09-01]

ISO/SAE 21434 section 9 describes the need of a clear definition of the item under consideration or also called Target of Evaluation (ToE). This item definition shall define the item, its operational environment and the interactions in the context of cyber security. As this paper aims to define security measures for an individual ECU, some assumptions regarding the items functionality and operational environment need to be taken in order to define detailed overview of its functionalities, rather than a generic description. The standard requires the identification of the following information regarding the ToE.

- item boundary
- item functions
- preliminary architecture

4.1.1. Item Boundary and preliminary architecture

ISO/SAE 21434 9.3.2 Requirements and recommendations note that an item boundary is necessary to distinguish the item from its operational environment. This description can include interfaces, connections to other components of the vehicle or vehicle external E/E systems. It also provides some information about the environment, context and scope of the ToE.
The first assumption in regard to the item definition is the definition of the E/E-Architecture. This assumption is necessary, as the placement and interconnection of electric/electronic components becomes increasingly important due to the rising connectedness and complexity of components and functionalities. [93] Compared to a traditional E/E-Architecture where ECUs tend to be directly connected to each other and sensors/actuators to enable the introduction of optional functionalities in all regards. An example of such an architecture can be found in figure 2.1. Components are clustered in application areas and connected to a central gateway which is responsible for the translation of different communication technologies. The gateways is therefore a complex component with its main purpose in communication

routing without providing much more functionalities than diagnosis functionality. As basically all communication runs through the gateway, it is always under load and with increased vehicle functionalities which require even more information exchange, this structure reaches its limitations and becomes a bottleneck due to too much traffic passing through.
Therefore, the first assumption and simultaneously proposal suggests the use of a domain-based architecture as depicted in figure 4.1.

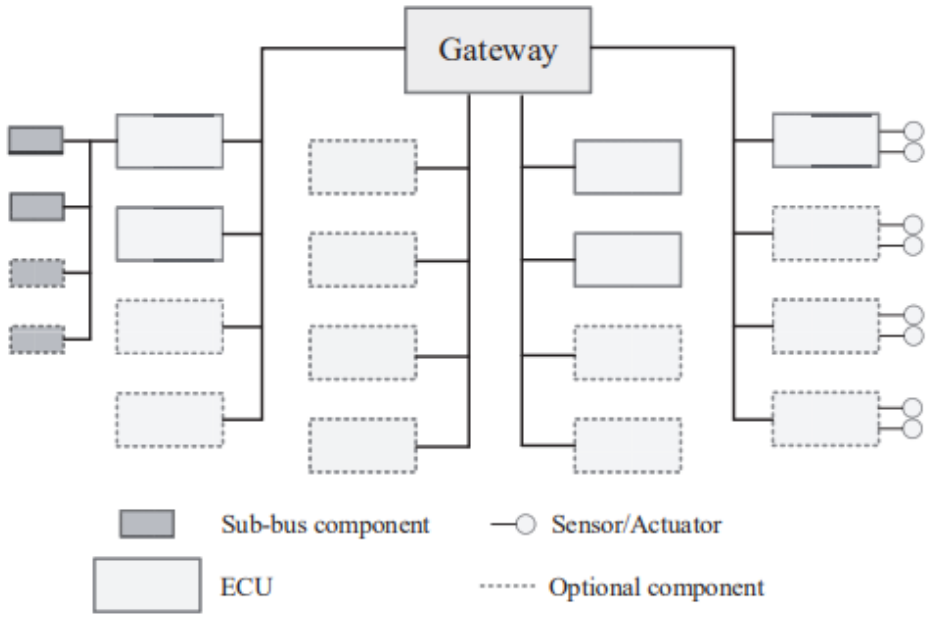

Figure 4.1.: Domain-based E/E-Architecture ([93])

Such an approach simplifies the structure by clustering the components into functional domains by integrating a high-performance domain controller component per domain. They are responsible for the communication within the domain and sole interface to the overall gateway. This change has great benefits as it reduces communication between ECUs significantly because most data exchange occurs within the functional domain. This in term also leads to reduced workload of the gateway itself. It also has benefits on the scalability of the system as it makes integration of added functionality easier due to the separation in domains. It also has cyber security benefits as the domain controllers can act the same way as network zones do in traditional IT systems and can act as firewalls and network segmentation devices. [94] However, new functionalities such as self-driving modes indicate that functionalities in the vehicle may overlap and further improvements to achieve the required timing constraints and sufficient quality of service are necessary.

According to ISO/SAE 21434 9.3.2 [RQ-09-01] the information about the item boundary and preliminary

architecture can be found from an architectural- and technical point of view in graphical form in figures 4.2 and 4.3 respectively.

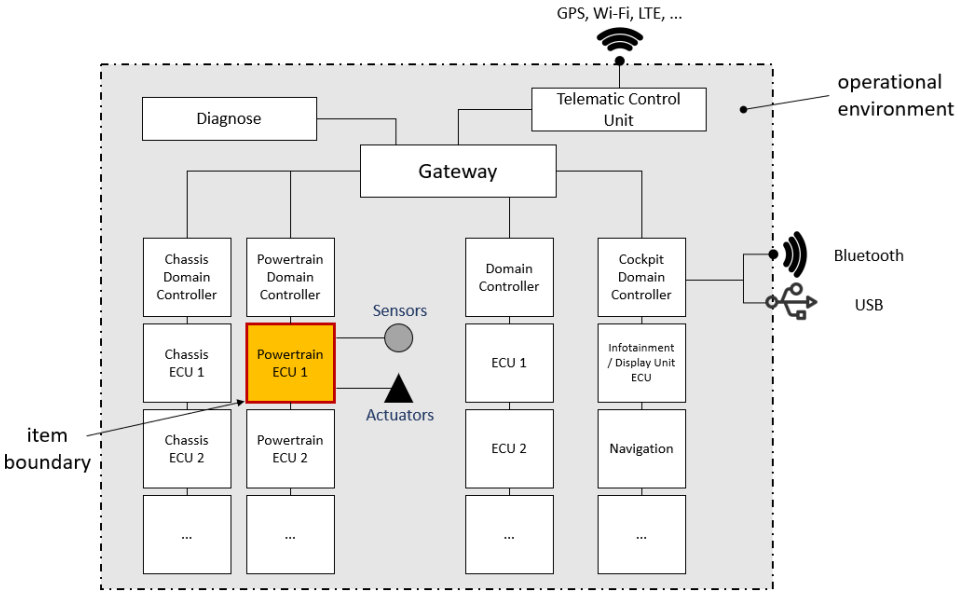

Figure 4.2.: Architectural preliminary architecture

As illustrated in figure 4.2 in yellow and red color, the Target of Evaluation is an individual ECU within the powertrain domain. The analysis will therefore focus on such an ECU with all its components including hardware, software and interfaces to other systems.

The powertrain domain has been chosen as the domain of interest as it has strict safety demands and real-time requirements and is therefore highly security relevant and can be considered as an integral part of the vehicle infrastructure. It is responsible for controlling the tractive effort to the wheels [95]. The powertrain is an assembly of all components which thrust the vehicle into motion consisting of a power source commonly a combustion engine, hybrid or electronic system, a transmission and driveshaft, axles and differential which are collectively referred to as the powertrain system which are controlled by the driver or computers in the case of self-driving vehicles. Common functions of the powertrain domain are summarized below:

- Control combustion
- Manage engine speed
- Exhaust system control
- Transmission functionalities
- Gearbox control

- Energy management

Based on figure 4.2 and in accordance with ISO/SAE 21434 [RQ-09-02], the operational environment of the item is described providing supplemental security relevant information for the activities of the TARA.

ID	Description of the operational environment	Assumptions
A1	The \gls{ToE} is connected to the Powertrain Domain Controller which handles data communication to the rest of the network.	The Powertrain Domain Controller has a firewall to prevent invalid data communication from external interfaces.
	The \gls{ToE} is directly connected to other control units within the Powertrain Domain.	
	The \gls{ToE} is connected via analoge and digital connections to sensors and actuators via hardware pins and facilitates input/output operations.	
A2	The \gls{ToE} does not have external communication interfaces.	The Powertrain Domain network is separated from the rest of the vehicle network and communication can only through the domain controller.
	The Gateway ECU is connected to external communcation interfaces. (OBD-II diagnosis port)	
A3	External communication (except diagnosis via OBD-II) is managed by the Telematic Control Unit.	The Telematic Control Unit has strong security controls including a firewall function.

Table 4.1.: Operational Environment Assumptions

After providing a description of the operational environment, architecture and domain, the item itself needs to be defined in more detail and a technical description including the functionalities of the system is depicted in figure 4.3.

The item (ECU2.1.1) is an embedded system to control a component of the powertrain system, for example the engine. When referring to functionalities during this description, the engine control unit will function as an example to provide a better understanding of the system by referring to concrete methods rather than generic terms. The ECU is responsible for determining certain vehicle functionalities like fuel injection and engine speed by reading input values from sensors within the vehicle or by evaluating and calculating different variables that are received from other control units. In the example of the engine control unit, these functionalities might be the air to fuel injection ratio, idle speed and valve timings for the engine to enable a optimized combustion. [96] To control such functionalities, input parameters are interpreted by the software in real-time to provide the processor with a complete picture of how the system is performing at a specific moment in time. Models and algorithms are used to optimize the operation of the vehicle by taking the input variables and generating the correct values and signals as outputs in order to control the operation of e.g. fuel injectors or engine speed. Simply put, in the example of the engine control unit, it takes the position of the accelerator pedal and various other sensor

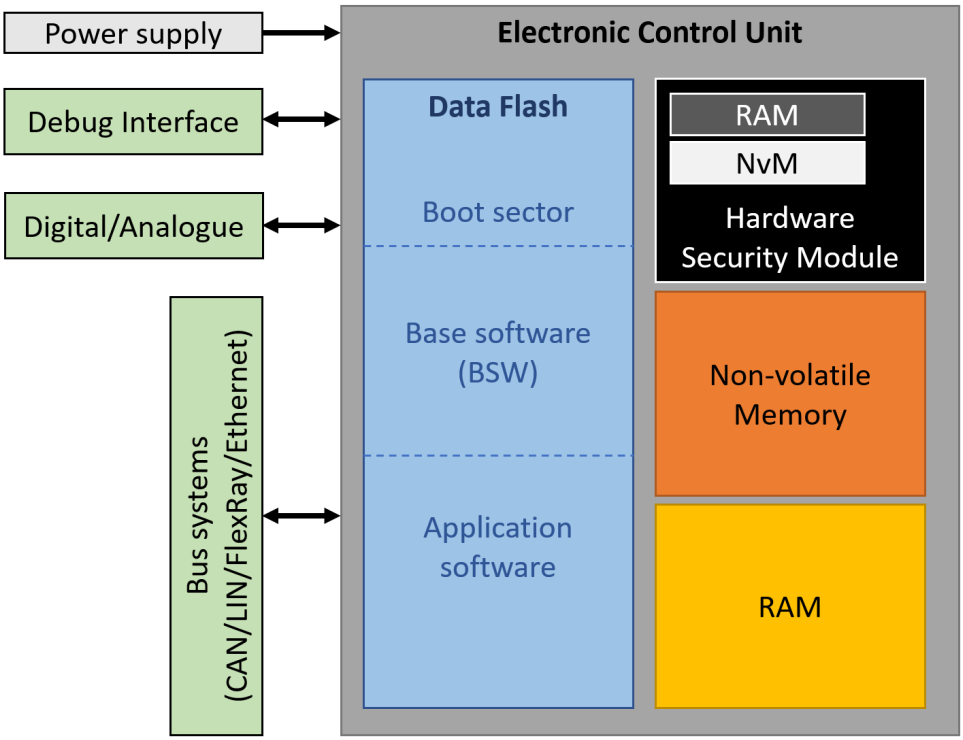

Figure 4.3.: Target of Evaluation (ECU)

inputs and calculates what values need to be changed in order to move the vehicle as the driver requested. It then transforms these values into signals which are furthermore sent to the hardware attached to the engine (like fuel injectors) and generates messages about the changes made based on the driver input to inform other control units which rely on this information.
The following list describes the different components and their use cases to get a better understanding of the Target of Evaluation.

Power supply provides the device with power from the battery when the vehicle is turned on

Debug Interface is used for development purposes and during production as well as for return analysis in case any issues need to be investigated after the SOP

Digital/Analogue signals are used to read/write data from/to sensors and actuators

Bus systems 2.1.4 are used for various purposes such as communication to other power-train systems, components within another domain and are used for uses like sharing information about system operation, provide diagnosis features and updates

Data Flash The data flash is the device internal memory of an embedded system used to store the software code (also referred to as program-flash or PFlash). As already described in the background

2.1.10, the software consists of BSW and application software. One more section of the flash is reserved for the boot operation of the device. This section is responsible to check the integrity and compatibility of the software and handles other features such as programming the device.

Hardware Security Module (HSM) 2.1.2 protects security relevant information from disclosure or alteration and provides efficient security functionalities. It has its own operating system and software which is executed on a secured processor with internal RAM and a secured NvM to store security relevant data.

Non-volatile Memory (NvM) acts as storage for data which needs to be persistently saved

Random Access Memory RAM executes the program code from PFlash

4.2. Asset identification [[RQ-15-02]]

This step of the analysis defines assets with cyber security properties whose compromise leads to a damage scenario. A complete overview of the identified assets and their related damage scneraios is provided in table 4.3 in the end of this section.
When considering the design of the ToE 4.3 some assets with cyber security properties can be identified based on the recommendations of J3061 [85]. The hardware security module is considered as a especially useful cyber security design option. The HSM and its functionalities are explained in section 2.1.2 and the software and functionalities provided by it need to be considered as assets in this analysis. The identified assets are summarized below:

- HSM software (Asset ID: A1)
- HSM content (Asset ID: A2)
- Trust anchors (Asset ID: A3)
- Cryptographic keys (Asset ID: A4)

4.2.1. Feature definitions

This section defines use cases for emerging vehicle technologies based on the literature analysis conducted in a previous chapters 2. The proposed approach will conducte the asset identification based on the structure of the literature analysis to systematically identify security relevant assets. As an analysis of each and every functionality of the vehicle is infeasible with available resources and time, the classification of different functionalities from table 2.3 is used for ADAS feature definitions. One feature of

each feature class will be selected to represent all features in that category. For the identification of assets regarding connectivity and communication 2.2.2 technologies the threat, vulnerabilities and risk analysis from the European Telecommunications Standards Institute for a ITS will be used. Since a different scope is used in their analysis, the results need to be adapted to fit the ToE 4.3.

Driving stability

As no ADAS features were identified during the literature analysis under the classification Driving Stability, the feature definition for this classification will be omitted as common functionalities such as traction control or ABS can be considered as proven in use and normal operation when it comes to ECU functionality.

Longitudinal Control

For the longitudinal control class, the Adaptive Cruise Control (ACC) feature has been chosen as the feature has stringent safety requirements and involves input, internal processes and needs to manage vehicle speed in real-time. [34] ACC is an advancement of cruise control functionality which dynamically adjusts the vehicle speed according to the current relative distance to the leading vehicle.
In order to provide this functionality, many components need to be involved and interact with each other. An analysis of such a system has already been conducted by research scientist Meiyuan Zhao [34] including a graphical illustration about the system architecture required to enable ACC functionality.

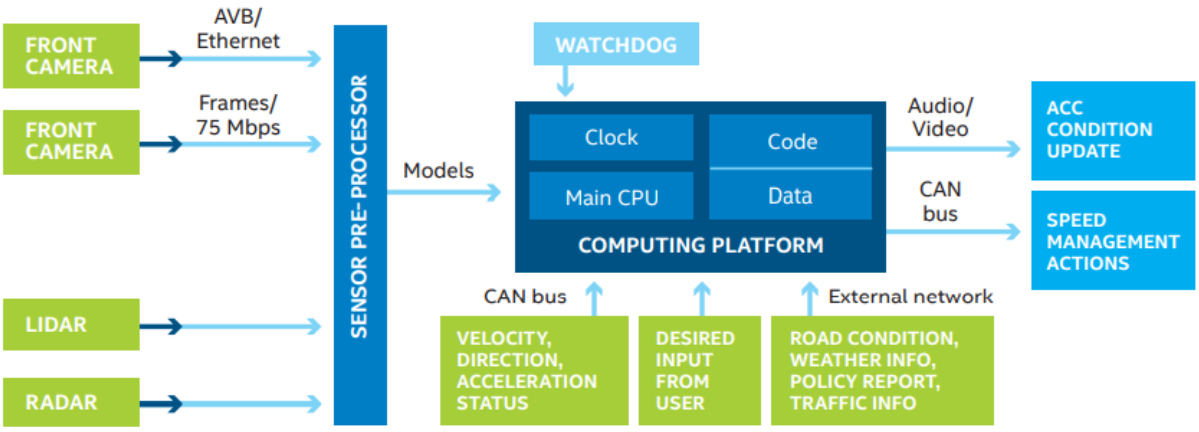

Figure 4.4.: ACC example system architecture ([34])

Figure 4.4 provides an overview of the system. The chosen approach aims to define threats for the target of evaluation 4.3 only, so only parts of the system will be considered which resemble the functionality of an ECU. In this system, the sensor pre-processor, computing platform and speed management system can

be considered as components with properties of an ECU. All three of these components share the same basic functionalities considering the workflow in figure 4.4 which can be generalized and the involved assets can be identified.

- obtaining input data from various sensors (Asset ID: A5)
- receiving bus signals and communication messages (Asset ID: A6)
- process data by executing software (Asset ID: A7)
- generating output data (Asset ID: A8)

Lateral Control

The lateral control feature representative of this category is the Lane departure Warning (LDW) feature which enables the vehicle to sense and calculate if the vehicle is driving properly in the lane. A warning is generated to warn the driver if the system determines that the vehicle is not moving properly in the lane or irregularly departing the lane. This feature involves similar technologies as the Adaptive Cruise Control feature and uses input captured from cameras, sensor input like vehicle speed and steering positon via messages and processes this data. In case the system determines dangerous behaviour, it generates output signals to warn the driver. The evaluation of such a system was also part of the analysis by Meiyuan Zhao [34].

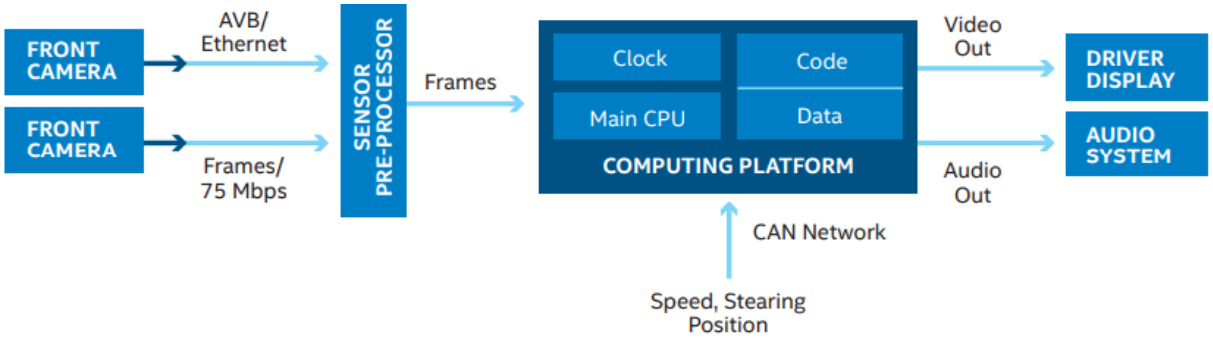

Figure 4.5.: LDW example system architecture ([34])

The architecture of figure 4.5 closely resembles the architecture of the longitudinal control functionality described in section 4.2.1. Therefore the assets involved which can be considered as relevant for this analysis are exactly the same.

- obtaining input data from various sensors (Asset ID: A5)
- receiving bus signals and communication messages (Asset ID: A6)

- process data by executing software (Asset ID: A7)
- generating output data (Asset ID: A8)

Cockpit

The only identified ADAS feature relevant for the cockpit category in table 2.3 was defined as the Indoor monitoring feature. Such a feature takes various inputs from sensors like cameras observing the driver, data from smart devices such as the current heart rate, steering behaviour or other relevant information to determine the current awareness state of the driver and warns the driver in case it detects fatigue or micro-sleep. Following the approach of the previous feature definitions, an example illustration of an architecture of such a system can be found in figure 4.6.

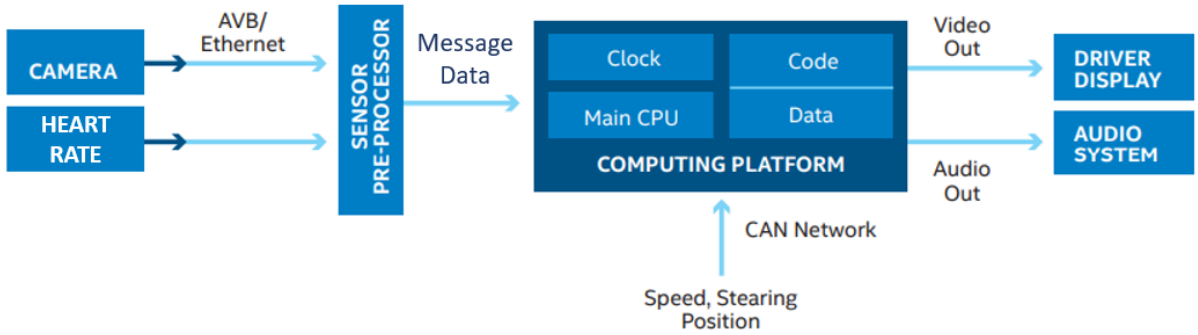

Figure 4.6.: cockpit example system architecture (modified graphic from LDW example [34])

As this feature also aims to provide a warning message to the driver, the general workflow and the architecture are quite similar to the later control feature depicted in figure 4.6. The assets involved which can be considered as relevant for this analysis are exactly the same.

- obtaining input data from various sensors (Asset ID: A5)
- receiving bus signals and communication messages (Asset ID: A6)
- process data by executing software (Asset ID: A7)
- generating output data (Asset ID: A8)

Parking

Automated Parking Assist is a feature which assists drivers during their park maneuver. Systems providing such a feature can detect parking spots which are big enough to fit the vehicle and notify the driver. Furthermore, they are able to change the steering wheel position based on sensor inputs to maneuver the

vehicle into the parking space. Most of the current systems only manage the steering mechanism, while the driver has to do the accelerating and breaking. [97] However, future applications include remote park assist by commanding the vehicle via smartphone or completely automated parking garages, which provide automated valet parking. Such a feature requires a lot of interactions with various modules of the vehicle in order to detect suitable parking spaces, calculating the optimal path into the parking space and necessary steering movements and number of maneuvers required. Such a system combines longitudinal with lateral control and involves automated steering and communication with the driver. An example illustration of an architecture for such a system is depicted in figure 2.2.1

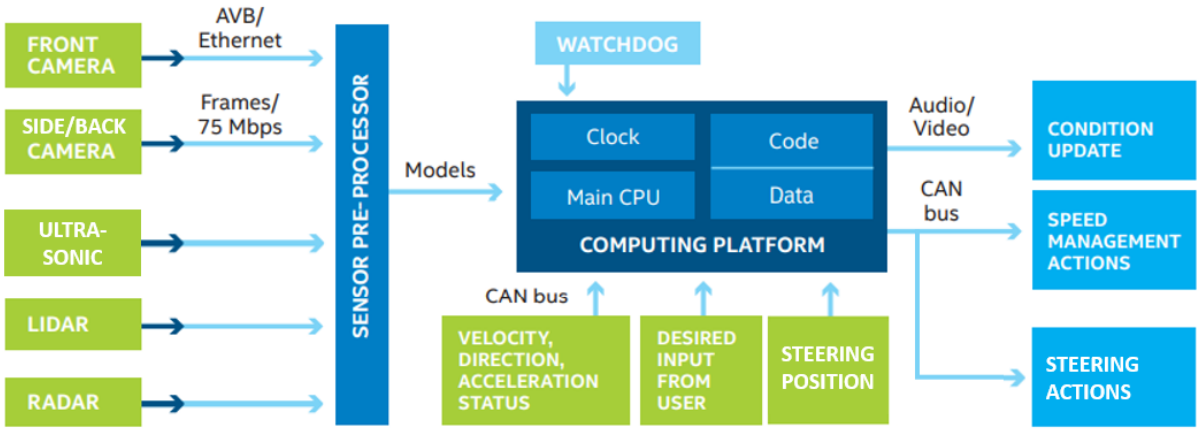

Figure 4.7.: Parking example system architecture (modified graphic from ACC example [34])

The combination of longitudinal, lateral and driver communication requires a more complex system with additional interactions between different components and ECUs. However, when considering the ToE, the asset identification results in the same outcome as previously assessed driver assistance features.

- obtaining input data from various sensors (Asset ID: A5)
- receiving bus signals and communication messages (Asset ID: A6)
- process data by executing software (Asset ID: A7)
- generating output data (Asset ID: A8)

Light and Sight

According to Ma [98] an adaptive front-lighting system is a lighting device, providing beams with automatic adaptation to varying conditions. This feature illuminates the road brighter or activates the high-beam during high speed and can deactivate these lights in case a vehicle is approaching. Other

applications include a wider beam in an urban environments for better pedestrian awareness, during cornering or illumination of the edges of the road during bad weather or strong lighting conditions due to oncoming traffic. Figure 4.8 depicts the system architecture of such a system based on the reference design of Ma [98].

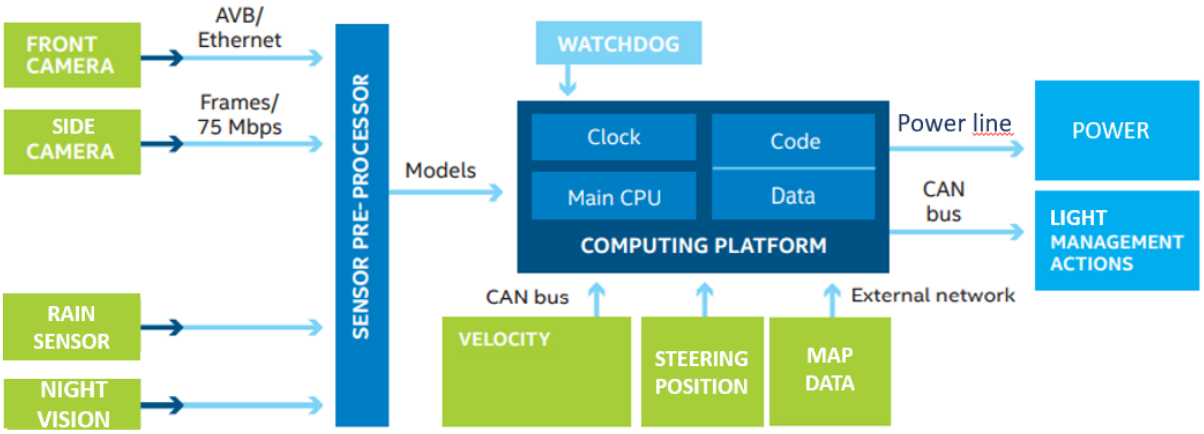

Figure 4.8.: Lighting example system architecture (modified graphic [34])

The asset identification of the selected use case within the light and sight feature category of ADAS shares the same identified assets as the rest of the categories.

- obtaining input data from various sensors (Asset ID: A5)
- receiving bus signals and communication messages (Asset ID: A6)
- process data by executing software (Asset ID: A7)
- generating output data (Asset ID: A8)

The analyzed use cases conclude the evaluation of the assets involved in ADAS features. The takeaway of this analysis is that even though features with completely different areas of application were analyzed, the assets regarding the ToE are consistent over all feature categories.

Connectivity and communication

The European Telecommunications Standards Institute has published a technical report regarding security threats, vulnerabilities and risks for Intelligent Transport Systems ITSs. [99] The scope of this report considers vehicle-to-vehicle and vehicle-to-roadside network infrastructure communication services assuming a fully deployed ITS. The TVRA method used in the report identifies an inventory of assets based on the definition of its ToE. The scope of the report is considerably bigger and resembles the operational

environment in our defined item boundary 4.2. Therefore this analysis will reduce the analysis to the items relevant to the ToE defined in this work 4.3.

The scope of the ETSI report is illustrated in figure 4.9. The functional and data assets from this graphic are explained in the following list:

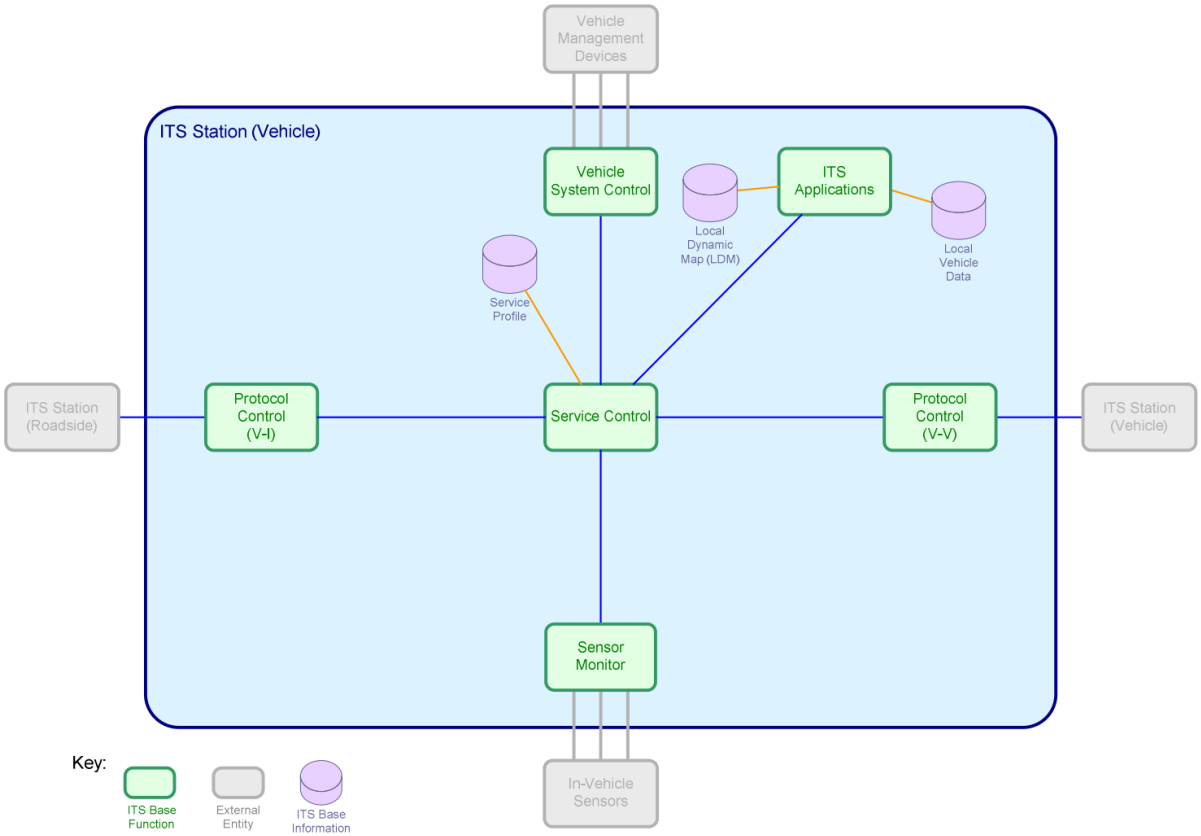

Figure 4.9.: ITS in-vehicle assets [99]

Protocol Control is responsible to select an appropriate message transfer protocol for external communication

ITS applications implement the functionalities provided by ITS. This includes the processing of ITS data for local use, maintaining information about the vehicle and its environment and initiating communication to internal (instruct the Service control component to take direct driving actions such as steering, accelerating or breaking) and external components (sending messages to other ITS participants)

Service Control enables the information exchange between other functional assets and manages inter-process communication

Sensor Monitor provides vehicle-internal environmental data for distribution to other functional assets

Vehicle system control allows ITS components to control vehicle behavior by instructing vehicle internal systems like the engine control unit to change speed

Vehicle management devices resemble the ToE 4.3 in this thesis and refers to components which directly control vehicle functionalities

Local Dynamic Map (LDM) consists of a dynamically updated data repository specific to the ITS application relating to local driving conditions including data from on-board sensors, on-board messages and vehicle external data

Local Vehicle Information includes data relating to the vehicle that is not relevant for real-time drving decisions and therefore not included in the LDM (e.g. manufacturing date, ...)

Service Profile includes the current profile of the list of ITS relevant applications installed, activated or disabled

Based on the assets defined in the technical report by ETSI [99], the assets relevant for the ToE 4.3 of this thesis can be identified:

- obtaining input data from various sensors (Asset ID: A5)
- receiving bus signals and communication messages (Asset ID: A6)
- process data by executing software (Asset ID: A7)
- generating output data (Asset ID: A8)
- configuration data (Asset ID: A9)

Based on the requirement within the ISO standard [RQ-15-02] and the defined architecture of the ToE 4.3 and identified features a complete overview including a description of the identified assets is provided in table 4.2.
Furthermore, ISO/SAE 21434 [RQ-15-01] calls for an identification of damage scenarios which are relevant for the identified assets. These damage scenarios are outlined in table 4.3.

ID	Asset	Description
A1	HSM software	This asset consists of the software and functionalities provided by the Hardware Security Module (Crypto Algorithms, Random Number Generation, …)
A2	HSM content	This asset refers to permanently stored data in the NvM of the Hardware Security Module (Log entries, certificates, ...)
A3	Trust anchors	This asset contains security relevant material used to authenticate external sources (root certificates, …)
A4	Cryptographic keys	This asset refers to cryptograhic key material used for encryption/decryption (symmetric and asymmetric keys, ...)
A5	Sensor data	This asset describes data that is sent/received to/from sensors as signals (digital, analogue signals)
A6	Message data	This asset consists of messages sent/received via various bus systems (CAN messages, …)
A7	ECU software	This asset refers to the software programmed to the ECU internal Flash (firmware)
A8	ECU functionality	This asset refers to the functionality provided by the ECU (engine control, communication, …)
A9	Configuration data	This asset consists of vehicle specific configuration data which is used by the software (drive type, active features, …)

Table 4.2.: Asset description

4.3. Damage scenario impact rating [RQ-15-04]

During this step of the analysis, the identified damage scenarios are assessed against the potential consequences for traffic participants. The impact categories are defined as proposed in ISO/SAE 21434:

S Safety

F Financial

O Operational

P Privacy

These impact categories are then furthermore detailed by an impact rating of a damage scenario:

S Severe

Ma Major

Mo moderate

N Negligible

Based on ISO/SAE 21434, the criteria for the impact rating for a given damage scenario is defined for all categories (safety, financial, operational and privacy).

ID	Asset	Security property			Damage scenario
		C	I	A	
A1	HSM software	X			Intellectual property is stolen by disclosure of software - Reputation damage for OEM
A1	HSM software		X		Modification of HSM software can lead to unexpected system behaviour
A2	HSM content	X			Readout of HSM content leads to unexpected behaviour of the ECU, but without safety impact as safety relevant functions are not impacted
A2	HSM content		X		Modification of HSM content can lead to unexpected behaviour of the ECU
A3	Trust anchors		X		Modification of trust anchors leads to unexpected behaviour of the ECU, but without safety impact as safety relevant functions are not impacted
A4	Cryptographic keys	X			Disclosure of cryptographic material can lead to unexpected system behaviour
A4	Cryptographic keys		X		Modification of cryptographic material can lead to unexpected system behaviour
A5	Sensor data	X			Intellectual property is stolen by disclosure of sensor data
A5	Sensor data		X		Modification of sensor data can lead to unexpected system behaviour
A6	Message data	X			Intellectual property is stolen by disclosure of message data
A6	Message data		X		Modification of safety relevant message data can lead to unexpected system behaviour
A7	ECU software	X			Intellectual property is stolen by disclosure of software - Reputation damage for OEM
A7	ECU software		X		Modification of ECU software can lead to unexpected system behaviour
A8	ECU functionality		X		Modification of ECU software can lead to unexpected system behaviour
A8	ECU functionality			X	System becomes unusable
A9	Configuration data	X			Intellectual property is stolen by disclosure of configuration data
A9	Configuration data		X		Modification of configuration data can lead to unexpected system behaviour

Table 4.3.: Asset identification [RQ-15-02]

Impact rating for safety damage

ISO/SAE 21434 [RQ-15-06] states, that safety related impact ratings shall be derived from ISO 26262-3:2018, 6.4.3. [92]

Severe S3: Life-threatening injuries (survival uncertain), fatal injuries

Major S2: Severe and life-threatening injuries (survival probable)

Moderate S1: Light and moderate injuries

Negligible S0: No injuries

Impact rating for financial damage

Severe The financial damage leads to catastrophic consequences which the affected vehicle manufacturer might not overcome.

Major The financial damage leads to substantial consequences which the affected vehicle manufacturer will be able to overcome.

Moderate The financial damage leads to inconvenient consequences which the affected vehicle manufacturer will be able to overcome easily.

Negligible The financial damage leads to no effect, is irrelevant or negligible. consequences

Impact rating for operational damage

Severe The operational damage leads to the permanent loss or impairment of a core vehicle function.

Major The operational damage leads to the permanent loss or impairment of an important vehicle function.

Moderate The operational damage leads to partial degradation of a vehicle function.

Negligible The operational damage leads to no impairment or non-perceivable impairment of a vehicle function.

Impact rating for privacy damage

Personally identifiable information (PII) and the definition of a PII principal is taken from ISO/IEC29100. PII is defined as "any information that (a) can be used to identify the PII principal to whom such information relates, or (b) is or might be directly or indirectly linked to a PII principal"
PII principal is defined as "natural person to whom the personally identifiable information (PII) relates" [100]

Severe The privacy damage leads to significant or even irreversible impact to the road user. The information regarding the road user is highly sensitive and easy to link to a PII principal.

Major The privacy damage leads to serious impact to the road user. The information regarding the road user is highly sensitive but difficult to link to a PII principal or sensitive and easy to link to a PII principal.

Moderate The privacy damage leads to inconvenient consequences to the road user. The information regarding the road user is either sensitive but difficult to link to a PII or not sensitive but easy to link to a PII principal.

Negligible The privacy damage leads to no effect or, negligible consequences or is irrelevant to the road user. The information regarding the road user is not sensitive and difficult to link to a PII principal.

4.3.1. Impact assessment

Based on the identified damage scenarios in table 4.3, the impact ratings for these damage scenarios are outlined in table 4.4.

Damage scenario	Damage scenario	Impact category	Impact rating
DS-1	Intellectual property is stolen	P	Mo
DS-2	Modification of asset can lead to unexpected system behaviour	S	S3
DS-3	System becomes unusable	O	Mo
DS-4	Reputation damage for OEM	F	Ma
DS-5	Disclosure can lead to unexpected system behaviour without safety impact	O	Mo
DS-6	Privacy impact for customer	P	Mo

Table 4.4.: Impact rating [RQ-15-04]

4.4. Threat scenario identification [RQ-15-03]

The workflow of ISO/SAE 21434 [RQ-15-03] requires that threat scenarios shall be identified and include the following details:

- Targeted Asset
- Compromised cyber security property of the asset
- Cause of compromise of the cyber security property

During this assessment, it has to be noted, that a damage scenario can correspond to multiple threats and a threat scenario can lead to multiple damage scenarios. Furthermore, the method for threat scenario identification uses a systematic approach by an elicitation of malicious use cases which are the result from reasonably foreseeable abuse and/or misuse based on the asset identification of use cases for the emerging vehicle technologies identified during the literature analysis 2. Table 4.5 shows the identified

threat scenarios leading to each damage scenario. Each threat can be identified by an incremental ID. Threat scenarios which lead to identical threats are omitted during the threat scenario identification but will be considered due to the different damage scenarios assigned to them.

4.5. Attack path analysis [RQ-15-08]

The ISO/SAE 21434 standard defines in [RQ-15-08] that all threat scenarios have to be analysed to identify potential attack paths. The chosen method, also proposed in the standard, uses a top-down approach in the form of an attack tree analysis. 2.4.1 All the identified attack paths are depicted in Appendix A. To provide a better understanding of the analysis, the first threat scenario will be explained in detail in this section. The illustrated threat scenario is relevant for two different damage scenarios (DS-1 and DS-4 in reference to the impact rating table 4.4) as depicted in figure 4.10 above the threat scenario in red color.

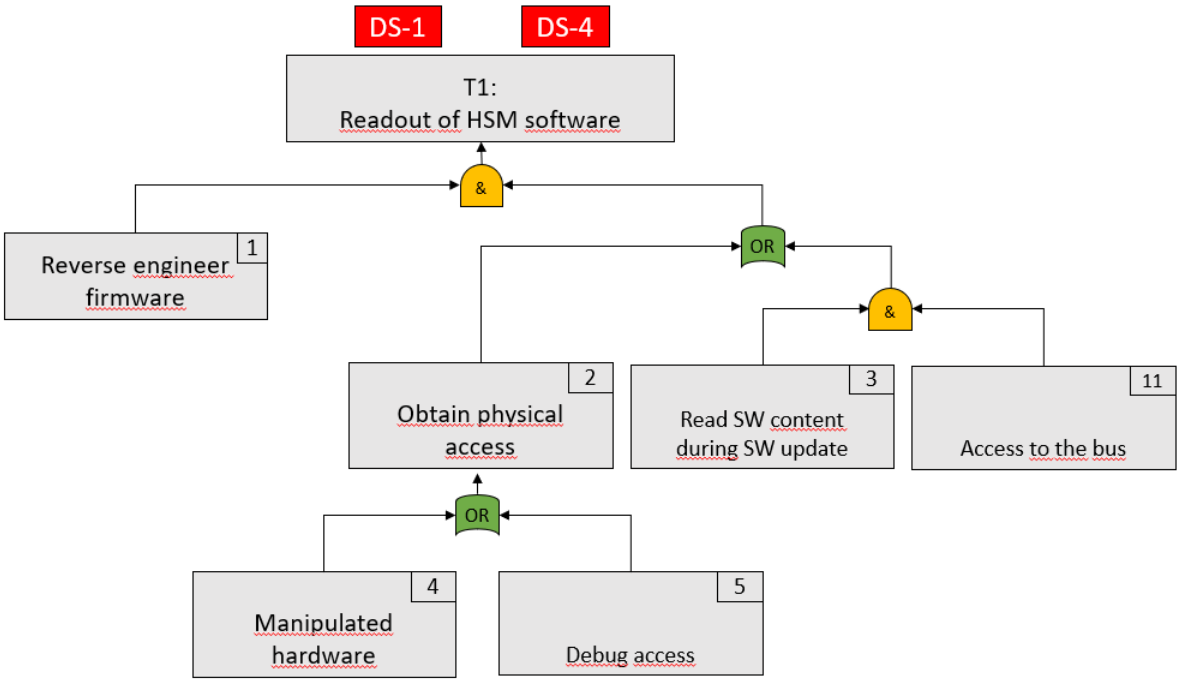

Figure 4.10.: Threat scenario 1

The threat scenarios referenced in table 4.5 are represented by their ID (T1). In this scenario, the overall goal of disclosing the HSM software can only be achieved by reverse engineering the firmware and successfully extracting the software content from the ECU. The extraction of the SW in this case is either possible by obtaining physical access or extracting the information from the bus during a software update. Each attack step is furthermore identifiable by an assigned ID in the top right of each step. In

some cases where a threat scenario is achieved by exploiting another threat scenario, the corresponding sub-goals are identified by their threat scenario ID (e.g. T6). These IDs will be used in the next chapter to determine the attack feasibility of each attack path and furthermore calculating the feasibility of each individual threat scenario by evaluating all attack paths.

The identified attack steps are presented in table 4.6. Additionally, the table contains information which attack step is part of a threat scenario and shows how often the attack step is overall used to achieve an attack goal. Notably, some attack steps occur within other threat scenarios which are used to exploit another attack goal and therefore occure multiple times within a single threat scenario.

ID	Damage scenario	Threat scenario	Threat ID	Asset ID	Property		
					C	I	A
DS-1	Intellectual property is stolen	Readout of HSM software leads to loss of HSM code confidentiality which can lead to losing competetive advantages against other manufacturers and helps hackers or car tuners to find vulnerabilities in program code to circumvent security	T1	A1	X		
		Readout of sensor data leads to loss of signal confidentiality which enables attackers to obtain information about the communication to the ECU	T2	A5	X		
		Readout of message data leads to loss of communication confidentiality which enables car tuners to obtain information about the communication to the ECU	T3	A6	X		
		Readout of ECU software leads to loss of firmware confidentiality which can lead to losing competetive advantages against other manufacturers and helps hackers or car tuners to find	T4	A7	X		
		Readout of configuration data leads to loss of configuration data confidentiality which can lead to losing competetive advantages against other manufacturers and enables car tuners to create custom configurations for specific vehicles	T5	A9	X		
DS-2	Modification of asset can lead to unexpected system behaviour	Modification of HSM software can lead to compromise of all security functionalities which enables an attacker to circumvent security mechanisms and impact safety critical functionalities.	T6	A1		X	
		Modification of HSM content can lead to compromise of all security functionalities which enables an attacker to circumvent security mechanisms and impact safety critical functionalities.	T7	A2		X	
		Disclosure of cryptographic keys material can lead to compromise of all security functionalities which enables an attacker to circumvent security mechanisms and impact safety	T8	A4		X	
		Modification of cryptographic keys material can lead to compromise of all security functionalities which enables an attacker to circumvent security mechanisms and impact safety	T9	A4		X	
		Modification of sensor data can lead to unexpected vehicle operation with safety impact.	T10	A5		X	
		Modification of message data can lead to unexpected vehicle operation with safety impact.	T11	A6		X	
		Modification of ECU software can lead to unexpected vehicle operation with safety impact.	T12	A7		X	
		Modification of ECU functionality can lead to unexpected vehicle operation with safety impact.	T13	A8		X	
		Modification of configuration data can lead to unexpected vehicle operation with safety impact.	T14	A9		X	
DS-3	System becomes unusable	Denial of Service attacks on sensor data can prevent the system from working properly and shutting down	T15	A5			X
		Denial of Service attacks on message data can prevent the system from working properly and shutting down	T16	A6			X
		Denial of Service attacks on the ECU functionality can prevent the system from working properly and shutting down	T17	A8			X
DS-4	Reputation damage for OEM	Readout of HSM software can cause negative media coverage for OEMs	-	A1	X		
		Readout of message data with privacy relevant information of customers can cause negative media coverage for OEMs	-	A6	X		
		Readout of ECU software can cause negative media coverage for OEMs	-	A7	X		
		Modification of ECU functionality can lead to unexpected vehicle operation with safety impact and therefore cause reputation damage for OEMs	-	A8	X		
DS-5	Disclosure can lead to unexpected system behaviour without safety impact	Readout of HSM content can lead to disclosure of security relevant information	T18	A2	X		
DS_6	Privacy impact for customer	Readout of message data with privacy relevant information of customers can cause data protection issues for OEMs	-	A6	X		

Table 4.5.: Threat scenario identification 1

ID	Attack step	Threat Ids	Occurences
1	Reverse engineer firmware	1,4,5	3
2	Obtain physical access	1,2,3,4,5,6,7,8,9,12,14	22
3	Read SW content during SW update	1,4,5	3
4	Manipulated hardware	1,2,3,4,5,6,7,8,9,12,14	22
5	Debug access	1,2,3,4,5,6,7,8,9,12,14	22
6	Intercept signal on bus	2,3	2
7	Obtain read access to ECU	2,3,4,5	4
8	Extract via diagnosis commands	2,3,4,5	4
9	Remote access	2,3,4,5,6,7,8,9	9
10	Software vulnerability	2,3,4,5,6,7,8,9,12,14	20
11	Access to the bus	1,2,3,4,5,6,7,8,9,10,11,12,14,15,16,17,18	28
12	Obtain write access to HSM	6,7,8,9	5
13	Read from bus during transmission	8	1
14	External disclosure (e.g. backend)	8	1
15	Modify during transmission	9	1
16	External modification (e.g. during production)	9	1
17	Modify or forge communication	10,11	4
18	Install rogue sensor	10	2
19	Obtain write access to ECU	12,14	12
20	Change boot mode	12,14	12
21	Exploit update procedure	12,14	12
22	External security issue (e.g. backend)	12,14	12
23	Interrupt communication	15,16,17	4
24	Exploit diagnosis commands (e.g. continuous reset)	17	1
25	Intercept HSM dump	18	1

Table 4.6.: Attack step overview

4.6. Attack feasibility assessment [RQ-15-10]

ISO/SAE 21434 presents multiple methods to determine the attack feasibility of each attack path, this paper assesses the attack feasibility rating with the attack potential-based approach. It is defined in ISO/IEC 18045 [101] as a measure of the effort which needs to be expended to successfully attack a component or item in terms of the expertise and resources required by the attacker. The method relies on five different components for this evaluation:

Elapsed Time as the time required to identify a vulnerability and develop an exploit 4.7

Specialist Expertise relates to the capabilities, skills and experience of an attacker 4.8

Knowledge of the system relates to the amount/classification of information about the system which an attacker needs to access 4.9

Window of Opportunity relates to the access which an attacker needs for a successful exploit like time available or physical/remote access 4.10

Equipment and tools which an attacker needs to execute the attack 4.11

4.6.1. Attack potential measure definitions

The following section defines the different measures for the attack-potential assessment taken from ISO/SAE 21434 which provides a customized definition from ISO/IEC 18045 [101].

Elapsed Time	Value
≤ one (1) day	0
≤ one (1) week	1
≤ one (1) month	4
≤ six (6) months	17
> six (6) months	19

Table 4.7.: Attack potential measure - Elapsed Time

Specialist Expertise	Description	Example	Value
Layman	No particular experience or expertisse, unknowledgeable compared to experts	Attacker using step-by-step descriptions of a publicly available attack method.	0
Proficient	Familiar with the security behaviour of the component or system type.	Technician or experienced owner knowing simple and popular attacks like chip tuning and installation of counterfeit parts.	3
Expert	Familiar with the underlying algorithms of the component or system (protocols, hardware, tools, principles and concepts of security employed, cryptography, classical attacks for the product type, attack methods)	Experienced engineer or technician.	6
Multiple experts	Expertise required to a great extend in different fields of expertise	Multiple experienced engineers who have expertise in various fields at an expert level for distinct steps of an attack.	8

Table 4.8.: Attack potential measure - Specialist Expertise

Knowledge of the system	Description	Example	Value
Public information	Public information about the component or system	Publicly available information in the internet	0
Restricted information	Restricted information about the component or system not publicly available	Internal documents shared between supplier and OEM (requirements, design specification)	3
Confidential information	Confidential information only available to specified teams	Source code or immobilizer-related information	7
Strictly confidential information	Strictly confidential information controlled on a need to know basis	Customer specific calibration data or memory maps	11

Table 4.9.: Attack potential measure - Knowledge of the system

Window of opportunity	Description	Example	Value
Unlimited	High availability via public network, access without time limitation	- Remote attack on server infrastructure - Owner has unlimited to OBD-II port	0
Easy	High availability with limited access time	- Attack while vehicle is standing still - Remote software update process	1
Moderate	- Low availability of the item or component - Limited physical/logical access - Physical access without any special tools	- Vehicle acces due to unlocked door resulting in physical access to OBD-II port	4
Difficult	Very low availability of the item or impractical level of access	- ECU needs to be removed from the vehicle - Strong cryptographic key is brute forced	10

Table 4.10.: Attack potential measure - Window of Opportunity

Equipment	Description	Example	Value
Standard	Equipment is broadly available or part of the product	Laptop, SW from manufacturer website, OBD-II adapter	0
Specialized	Equipment is not easily available but can be acquired or developed from legal sources (even though expensive)	Signal generator, specialized debugging or testing hardware/software	4
Bespoke	Equipment is specially produced, restricted or even illegal or immensly expensive	Manufacturer tools, electron microscope	7
Multiple bespoke	Different types of bespoke equipment are required for distinct steps of an attack.	Multiple tools of "Bespoke" category	9

Table 4.11.: Attack potential measure - Equipment

4.6.2. Attack feasibility mapping

The attack potential corresponds to the sum of all five attack potential measures which are then classified into the attack feasibility rating shown in table 4.12 consisting of four categories: High, Medium, Low and Very Low which are taken from ISO/IEC 18045 [101].

Attack feasibility rating	Value Range
High	0 - 9
	10 - 13
Medium	14 - 19
Low	20 - 24
Very Low	≥ 25

Table 4.12.: Attack feasibility mapping

All identified attack steps from the threat scenarios in A are analyzed and the attack potential of each attack path is calculated based on the values defined in 4.6. The aggregated values of each attack step are presented in table 4.13.

Based on these values, the threat scenarios in A are compartmentalized into different attack paths resulting into the goal of the attacker. Each intermediate attack step gets its attack feasibility based on the child nodes in the attack tree. If the steps are connected via AND the resulting attack feasibility is the sum of all child nodes and in case of an OR the attack path with the lowest feasibility is chosen. Based on this approach, the resulting attack feasibility of each threat scenario is illustrated in table 4.14

ID	Attack step	Attack feasibility assessment					
		Time	Expertise	Knowledge	Access	Equipment	Aggregate
1	Reverse engineer firmware	4	6	3	0	0	13
2	Obtain physical access	0	3	7	4	4	18
3	Read SW content during SW update	0	3	0	1	4	8
4	Manipulated hardware	19	8	11	10	9	57
5	Debug access	0	3	7	4	4	18
6	Intercept signal on bus	0	3	0	1	4	8
7	Obtain read access to ECU	0	3	0	4	4	11
8	Extract via diagnosis commands	0	3	0	0	4	7
9	Remote access	19	8	11	10	9	57
10	Software vulnerability	19	8	11	10	9	57
11	Access to the bus	0	0	0	1	0	1
12	Obtain write access to HSM	0	3	0	4	4	11
13	Read from bus during transmission	0	3	0	1	4	8
14	External disclosure (e.g. backend)	19	8	11	10	9	57
15	Modify during transmission	0	6	3	1	4	14
16	External modification (e.g. during production)	19	8	11	10	9	57
17	Modify or forge communication	0	3	3	1	0	7
18	Install rogue sensor	1	6	6	4	4	21
19	Obtain write access to ECU	0	3	0	4	4	11
20	Change boot mode	0	6	7	4	4	21
21	Exploit update procedure	19	8	11	10	9	57
22	External security issue (e.g. backend)	19	8	11	10	9	57
23	Interrupt communication	0	3	0	1	0	4
24	Exploit diagnosis commands (e.g. continuous reset)	0	3	0	1	0	4
25	Intercept HSM dump	0	6	7	10	4	27

Table 4.13.: Attack step feasibility

ID	Threat scenario	Attack feasibility assessment					
		Time	Expertise	Knowledge	Access	Equipment	Aggregate
T1	Readout of HSM software	4	6	3	1	4	18
T2	Readout of sensor data	0	3	0	0	4	7
T3	Readout of message data	0	3	0	0	4	7
T4	Readout of ECU software	4	6	3	0	4	17
T5	Readout of configuration data	4	6	3	0	4	17
T6	Modification of HSM software	0	3	7	4	4	18
T7	Modification of HSM content	0	3	7	4	4	18
T8	Disclosure of crypto material	0	3	0	1	4	8
T9	Modification of crypto material	0	6	3	1	4	14
T10	Modification of sensor data	0	3	3	1	0	7
T11	Modification of message data	0	3	3	1	0	7
T12	Modification of ECU software	0	3	7	4	4	18
T13	Modification of ECU functionality	0	3	3	1	0	7
T14	Modification of configuration data	0	3	7	4	4	18
T15	DoS on sensor data	0	3	0	1	0	4
T16	DoS on message data	0	3	0	1	0	4
T17	DoS on ECU functionality	0	3	0	1	0	4
T18	Readout of HSM content	0	3	7	4	4	18

Table 4.14.: Threat scenario feasibility

4.7. Risk evaluation [RQ-15-15]

The methodology described in ISO/SAE 21434 requires that a risk value shall be determined for each threat scenario A based on the attack feasibility and impact of the associated damage scenario 4.4. Such a risk value shall be a value between 1 and 5 with 1 representing the lowest risk. Both risk matrices or risk formulas may be used for this purpose.
In order to achieve such a value, the determined impact ratings and attack feasibility values need to be transformed into the appropriate format. The conversion for attack feasibility is based on table 4.6 whereas the translation to numerical values for both impact and attack feasibility can be found in table 4.15. The risk values are determined based on the following risk formula:
Risk = 1 + I × F

Impact rating	Numerical value I for impact
Negligible	0
Moderate	1
Major	1,5
Severe	2

Attack feasibility rating	Numerical value F for attack feasibility
Very Low	0
Low	1
Medium	1,5
High	2

Table 4.15.: Numerical value translation of impact and attack feasibility

Threat scenarios which correspond to multiple damage scenarios are evaluated according to the damage scenario with the highest impact rating. If the impact ratings are equal and one of the categories refers to safety, the safety relevant damage scenario is prioritized.
Table 4.16 shows the resulting risk values based on a calculation of attack feasibility (F) and impact rating (I) using the risk formula.

4.8. Risk treatment [RQ-15-17]

According to ISO/SAE 21434 [RQ-15-17] a risk treatment option shall be determined for each threat scenario. As this paper aims to determine all foreseeable security measures for ECUs, the risk treatment decision for all scenarios is defined as: **reducing the risk**.

ID	Threat scenario	Attack feasibility assessment			Impact Rating			Risk value
		Value	Rating		Scenario	Rating		
T1	Readout of HSM software	18	Medium	1,5	P, F	Major	1,5	3,25
T2	Readout of sensor data	7	High	2	P	Negligible	0	1
T3	Readout of message data	7	High	2	P, F	Major	1,5	4
T4	Readout of ECU software	17	Medium	1,5	P, F	Major	1,5	3,25
T5	Readout of configuration data	17	Medium	1,5	P	Negligible	0	1
T6	Modification of HSM software	18	Medium	1,5	S	Severe	2	4
T7	Modification of HSM content	18	Medium	1,5	S	Severe	2	4
T8	Disclosure of crypto material	8	High	2	S	Severe	2	5
T9	Modification of crypto material	14	Medium	1,5	S	Severe	2	4
T10	Modification of sensor data	7	High	2	S	Severe	2	5
T11	Modification of message data	7	High	2	S	Severe	2	5
T12	Modification of ECU software	18	Medium	1,5	S	Severe	2	4
T13	Modification of ECU functionality	7	High	2	S, F	Severe	2	5
T14	Modification of configuration data	18	Medium	1,5	S	Severe	2	4
T15	DoS on sensor data	4	High	2	O	Moderate	1	3
T16	DoS on message data	4	High	2	O	Moderate	1	3
T17	DoS on ECU functionality	4	High	2	O	Moderate	1	3
T18	Readout of HSM content	18	Medium	1,5	O	Negligible	0	1

Table 4.16.: Risk evaluation

4.9. Cyber security goals [RQ-09-05]

ISO/SAE 21434 [RQ-09-05] calls for the specification of cyber security goals for all threat scenarios where a reduction of the risk is necessary. The standard defines a cyber security goal as "a requirement to protect assets against a threat scenario" [13]. The approach in this paper will determine and assign the cyber security goals based on the individual attack steps defined in table 4.13. In order to identify each security goal, IDs will be assigned to each individual goal in the form of "SG-X".

Besides elicitation of security goals based on the attack step evaluation, security requirements and best practices in automotive standards like J3061 [85], UNECE WP.29 155e [87], EVITA [102], ETSI [99], PRESERVE [103], NHTSA best practices [104] and AUTO-ISAC [89] are considered. Although most requirements are not specifically defined for ECUs in these documents, the requirements for other components can be transposed to fit the ToE.

SG-1: Obfuscation techniques shall be used to make reverse engineering methods more difficult. [85]

SG-2: The ECU hardware shall be secured against unauthorized physical access. [99]

SG-3: Access control mechanisms to restrict access to critical ECUs shall be employed. [87]

SG-4: Confidential data transmitted to or from the ECU shall be protected. [99]

SG-5: The tampering of the HSM shall be prevented. [102]

SG-6: The ECU hardware shall be protected against fault attacks. [99]

SG-7: Development interfaces, communication channels and data shall be removed before production. [104]

SG-8: Unused hardware interfaces shall be permanently disabled. [103]

SG-9: Flash interfaces shall be protected by strong security mechanisms. [85]

SG-10: Confidential security artifacts shall only be stored within the HSM in encrypted form. EVITA [102]

SG-11: Access protection mechanisms shall be implemented to secure sensitive data. [85]

SG-12: Access control techniques and designs shall be applied to restrict access to critical modes of every ECU (diagnostic mode) [85]

SG-13: SW functionalities shall be protected against malicious misuse of diagnosis features. [104]

SG-14: Network segmentation in the form of isolation/partitioning of systems that have external access (e.g., Wi Fi, Bluetooth, OBD) from safety-critical systems shall be utilized. [85]

SG-15: Security controls shall be applied to systems that have remote access. [87]

SG-16: Secure coding standards shall be followed during SW development (as outined in NIST 8151 [105] or ISO/SAE 21434 [13])

SG-17: The ECU firmware shall be free of bugs and other errors. [105]

SG-18: Security relevant events shall be recorded. [104]

SG-19: Security incidents shall be recognized and reported for incident response. [85]

SG-20: Confidential security computations shall only be conducted within the secured memory of the HSM module. [87]

SG-21: Confidential security artifacts shall only be transmitted in encrypted form. [87]

SG-22: Security controls shall be implemented for storing cryptographic keys (e.g., use of HSM) [87]

SG-23: The ECU shall verify the authenticity and integrity of messages it receives. [87]

SG-24: ECU individual key material shall be generated within the security module of the ECU. [87]

SG-25: A True Random Number Generator (TRNG) shall be used to generate unique cryptographic key material.

SG-26: Spoofing of messages shall be prohibited. [87]

SG-27: A replay attack of messages shall be prohibited. [99] [102]

SG-28: The replacement of authorized electronic components (e.g. sensors) shall be prohibited / detected. [102]

SG-29: Logging data shall be protected against manipulation. [87]

SG-30: A manipulation of the SW shall be detected. [102]

SG-31: A secure boot mechanism shall be implemented. [104]

SG-32: Secure software update procedures shall be employed. [87]

SG-33: Only authorized SW shall be flashable. [103]

SG-34: Security controls for external systems shall be implemented. [87]

SG-35: In-vehicle network intrusion detection capabilities shall be implemented. [85]

SG-36: Protection mechanisms shall be in place restricting the usage of critical diagnosis commands under predefined circumstances. [104]

4.10. Countermeasure definition [RQ-09-08]

The following chapter defines countermeasures/cyber security controls based on the cyber security goals defined in section 4.9. The approach also considers countermeasures for the operational environment if the ToE is also impacted.

4.10.1. General security measures

Security goals covered in this chapter: SG-1, SG-3, SG-4, SG-7, SG-8, SG-9, SG-12, SG-15, SG-16, SG-17, SG-21, SG-34

Assumptions and general best practices regarding cyber security measures and implementations are covered in this section which do not suite specific functionalities furthermore detailed in this chapter. This

also includes areas outside of the item and operational environment like development activities or the backend.

Best practices

OWASP Top 10 A06:2021 - Vulnerable and Outdated components suggests that a process should be in place to remove unused or unnecessary components. [106] This best practice concerns SG-7 and SG-8 and is also noted in NHTSA [104] that developer-level access shall be eliminated if there is no operational reason. If such access is necessary, the interfaces should be appropriately protected and only accessible to authorized users. Only removing or hiding connectors and pins is not considered as a sufficient form of protection.

Secure coding principles and defensive implementation techniques need to be followed in order to produce quality software. [13]

Strong cryptographic algorithms, infrastructures, protocols, implementations and management functions shall be used according to FIPS/NIST or similar recommendations. [107]

Best practices for the protection of data integrity and confidentiality as defined in ISO/SC27/WG5 shall be followed for storing personal data. [87]

Obfuscation

Code obfuscation techniques could theoretically be used to make reverse engineering of firmware code more difficult, Gadina [108] discusses some methods how machine code may be obfuscated. However, obfuscation impacts performance, introduces even more complexity and does not prevent reverse engineering but only slows down the process. Therefore techniques which protect the confidentiality of firmware code should be preferred to security by obscurity techniques.

Public Key Infrastructure (PKI)

In order to provide suitable solutions for the security challenges faced in the automotive industry, a PKI needs to be in place for key and certificate management. Such a system needs to be in place for OEMs as well as suppliers and manufacturers in order to cover the need for secure distribution of cryptographic material both during production and in field operations. The difficulties in the automotive landscape originates from decentralized production facilities run by different companies in order to produce all components required for a modern vehicle. In order to overcome this challenge, companies

like ESCRYPT [109] follow a combined approach of a centralized and decentralized solution by using a centralized backend for key management and a decentralized provisioning system in production plants to achieve maximum availability, low latency and high security. Such a system can be used by manufacturers with a direct connection to third party or OEM backends to provision ECUs with a secure root of trust and to manage ECU individual key material. The establishment of such a system is essential for use cases where secured communication channels need to be established with external systems.
The need for ECU individual key material becomes imminent in recent attacks on vehicles such as an attack on a head unit by Subaru [110] shows. Any attacks which are able to obtain cryptographic material from a single vehicle or ECU shall not provide access to other components or vehicles.

Authentication

Providing the possibility for digital data exchange from the vehicle to external systems must depend on an appropriate authentication scheme based on cryptographic keys and certificates. The implementation and maintenance of such a system is challenging in a globally distributed environment and diverse supply chain structures like the automotive industry. [109]
A solution to this problem is the usage of certificate based authentication shown in figure 4.18.

Key management

Key and certificate management is relevant over the entire vehicle life cycle as depicted in figure 4.11. During production a critical step is the injection of the root certificate into the HSM module to establish a root of trust.

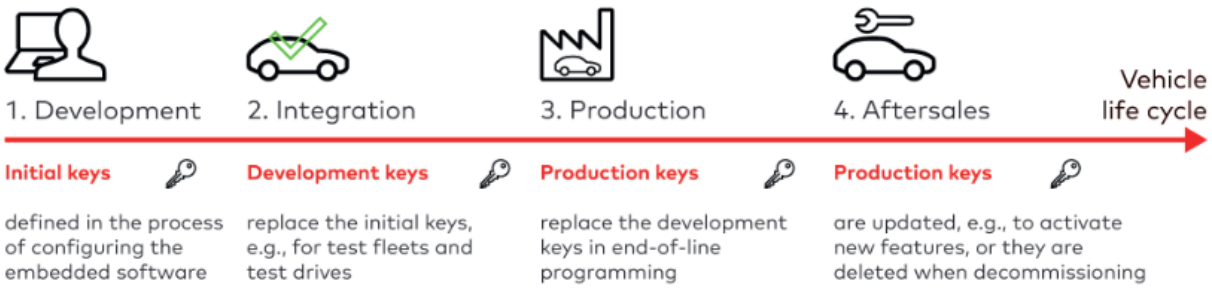

Figure 4.11.: Example HSM architecture [111]

Individual certificates which can be used to establish certificate based ECU authentication can be provisioned to ECUs by the usage of certificate signing requests (CSR). Hereby the HSM generates a key pair and includes the generated public key in a CSR data structure (e.g. including data like device id, feature set, ...). The request is then signed by the generated private key and the signature is attached to

the certificate. After that, the CSR is transmitted to the respective PKI where an identity certificate is created and written back into the ECU. [112]

Encryption

Encryption mechanisms are provided by the hardware protected security environment 4.10.2 cryptographic algorithms. The ECU may therefore use symmetric or asymmetric methods to encrypt the data which is sent to other network participants. However, CAN messages of real-time safety critical components can be protected against manipulation by the usage of SecOC 4.10.5 but encryption and decryption of messages is not considered feasible during operation. OEMs therefore need to evaluate what kind of data is confidential and needs to be protected against disclosure.

4.10.2. Hardware protected security environment

Security goals covered in this chapter: SG-5, SG-6, SG-10, SG-20, SG-22, SG-24, SG-25

A hardware protected security environment is required to protect ECUs. This environment is realized by using a HSM 2.1.2. The SAE standard J3101 [113] presents a common set of requirements which need to be implemented in such an environment which are represented in categories below:

Cryptographic key protection is referring to the protected usage and storage environment for the purpose of managing and storing keys also refered to as a "keystore". This overlaps with secure storage, but the J3101 standard specifically defines the keystore as distinct from the more general non-volatile data protection. This category also includes secure key usage by the utilization of an API enabling the outside environment to use the keystore securely.

Cryptographic algorithms are implemented within the hardware protected security environment to maintain confidentiality, integrity and availability to the HSM external environment. Such an implementation may use hardware accelerators in use cases where general purpose computing hardware does not satisfy system performance.

Random Number Generators are used for cryptographic purposes like key generation or challenge/response mechanisms. Properties of suitable cryptographic random bit generators are described in NIST SP-800-90B. [114]

Secure storage / non-volatile memory is used by cryptographic protocols and make use of retained data which needs to be protected against disclosure and manipulation. A storage area typically changes regularly and needs secure management. Example use cases for such a memory

region includes security relevant log entries, security parameters or reference values for manipulation detection.

Algorithm agility refers to the need of a flexible design in order to enable secured updates of the security environment withing losing or disclosing security relevant information.

Interface control needs to be implemented as the environment needs to be protected regarding its debugging and functional interfaces.

Secure execution environment refers to the usage of an isolated execution environment without the usage of untrusted resources. This isolated environment distinguishes the hardware protected security environment from the ECU environment.

Self-Tests evaluate the correct operation of critical security and operational functionalities and is required to ensure the integrity of the environment and safe and secure execution.

An example hardware design based on a closed on-chip realization within a ECU where application core and HSM are implemented on the same chip connected by a communication link is presented in figure 4.12 and discussed in further detail by Wolf [15].

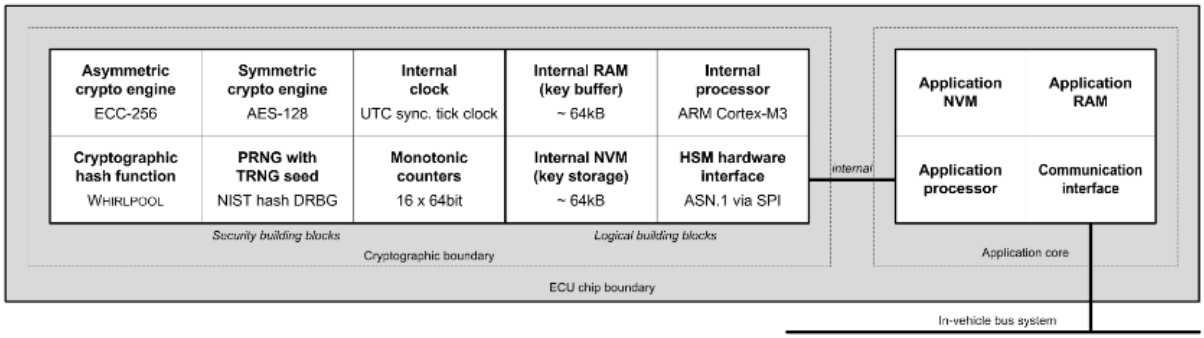

Figure 4.12.: Example HSM architecture [15]

Considerations

HSMs represent the root of trust of each ECU but increases the cost of each component and is not employable in all components such as light-weight sensors or actuators.

4.10.3. Manipulation/Intrusion detection system (IDS)

Security goals covered in this chapter: SG-14, SG-19, SG-30, SG-35

The ISO 21434 standard [13] defines an Incident Response process which requires OEMs to react to

security incidents that occur in their vehicles. In order to recognize such vulnerabilities and incidents, an IDS can be utilized. Such a system needs a distributed architecture as described in the AUTOSAR release R20-11 [115]. The proposed solution according to the standard consists of the following elements:

Security Sensors report security events to the IDSM.

Intrusion Detection System Manager (IDSM) is responsible to buffer reported security events and applies filters to qualify them and determine the criticality of events. Depending on the configuration of the system, these events can then be presisted on individual ECUs or forwarded to the IDSR.

Security Event Memory (SEM) acts as storage for security events

Intrusion Detection System Reporter (IDSR) receives qualified security events from other components and is responsible to enrich the received data with information and depending on the configuration forwarding critical security events to a Security Operation Center (SOC) or a Security Incident and Event Management (SIEM) solution.

Figure 4.13 illustrates the proposed architecture of a distributed vehicle IDS system.

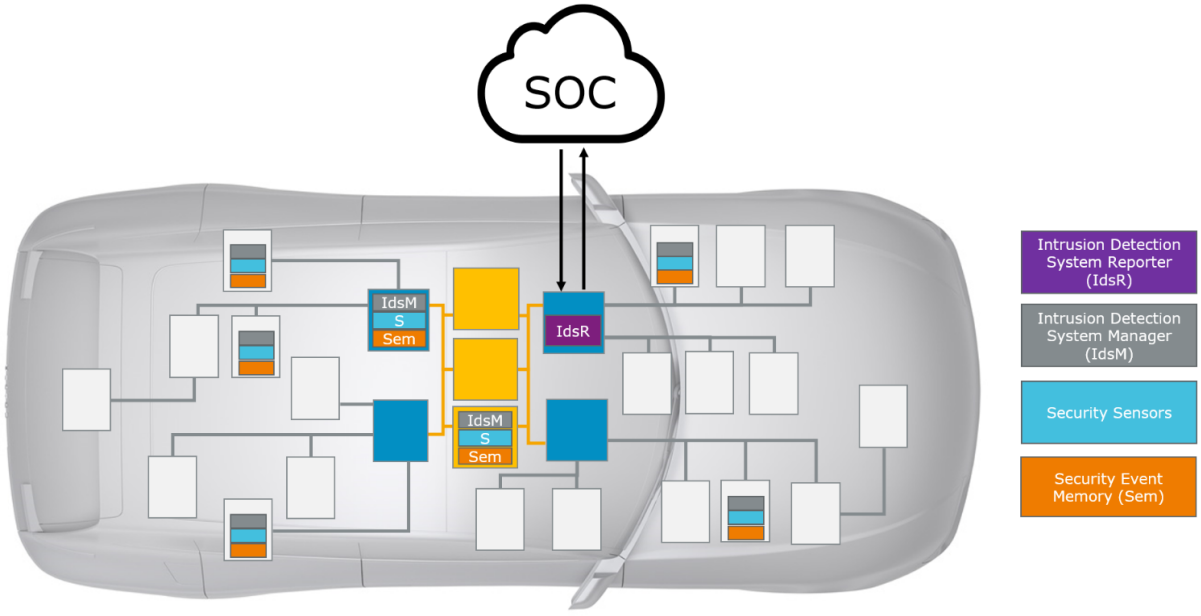

Figure 4.13.: IDS distributed system architecture [115]

When considering the vehicle architecture during the item definition 4.2 the telematics unit seems to be a suitable component to function as a IDSR module whereas ECUs can be considered as security sensors and IDSM modules. The security event memory acts similar to a secure logging 4.10.11 feature and

could be realized by utilizing the HSM secure storage 4.10.2.

Considerations

The introduction of an IDS is challenging in a heterogeneous environment like a vehicle network. Diverse ECUs from various manufacturers implement different soft- and hardware components which require a common set of functionalities to enable reliable reporting of security events. Furthermore, the introduction of such a system is subject to various security concerns itself such as trust between individual components and ensuring authenticity and integrity of message exchanges. Additionally, the configuration of reported security events and filtering unimportant events is a complex task with the risk of congesting the network due to additional traffic on the transport medium. [116]

4.10.4. Debugger Protection

Security goals covered in this chapter: SG-2

An interface for debugging purposes with access to the device processor is usually realized by the utilization of a Joint Test Action Group (JTAG) port. [117] The JTAG interface provides access to internal processor state by utilizing IEEE-1149 [118] standard instructions. By using this interface, an attacker could gain access to critical resources of the device including the threat of overwriting flash memory or reading and writing directly to processor registers or memory. Therefore it is of critical importance, that such an interface is protected against misuse.

To provide an appropriate security level to production ECUs, access modes can be configured to regulate access capabilities. It is important that such a measure is tamper-resistant which can be achieved by deriving the access possibilities from a set of fuses which are burnt and irreversibly increase the access protection. Such a mechanism can therefore only increase the security level but not reduce it. Different levels of protection levels can be configured up to a complete removal of the interfaces functionality in cases where it is more economical to replace the device instead of repairing it. However, it can be beneficial to have the possibility to access the interface for troubleshooting or return analysis. Therefore a strong authentication mechanism is required to protect the interface against misuse.

A strong user authorization is required to provide the required security level which requires a secured database to be maintained in order to store the credentials for device access. Due to the limited capabilities of embedded devices, such a functionality is often delegated to a trusted authority. Figure 4.14 provides an overview of a possible solution for an authorization system where the device provides a verification process (e.g. by requesting a random number from the HSM) which is then received by a

supporting software tool which needs to be connected to a secured infrastructure (such as a PKI) with a strong authentication mechanism (e.g. via Smart-Card). After solving the verification process by using the private key stored in the PKI, the JTAG interface can be unlocked to provide access only to trusted entities.

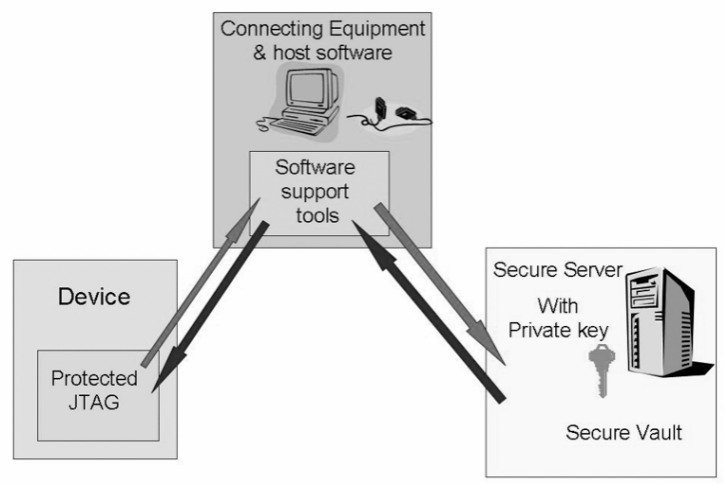

Figure 4.14.: JTAG authorization system [117]

Considerations

Based on the functionalities and hardware protected security environment 4.10.2 provided by the embedded system, the complexity of the verification process might be limited.

4.10.5. Secure Onboard Communication SecOC

Security goals covered in this chapter: SG-23, SG-26, SG-27

SecOC is an AUTOSAR module which aims to provide Protocol Data Unit PDU (message) integrity and authentication. [119] Integrity and authentication is necessary to protect correct and safe functionality of automotive vehicles as it ensures that data comes from the correct ECU and has the right value. The module aims for resource efficiency and practical authentication mechanisms of sensitive PDU data. The proposed approach supports both symmetric and asymmetric methods with their benefits and drawbacks. In order to ease the presentation and readability of this feature, the following explanation of the system will detail the symmetric approach and cover differences to the asymmetric approach in the considerations.

The AUTOSAR approach provides functionalities to verify the freshness and authenticity of communication between ECUs within the vehicle network. Both ECUs need to implement the SecOC module as

defined in the standard and provide interfaces to the communication module of the ECU also referred to as PDU router. The message freshness to protect the communication is provided by an external Freshness Manager in the form of either an incremental counter or secured time protocol by using timestamps.
The protocol uses two distinct PDUs to protect communication against unauthorized manipulation and replay attacks:

Authentic PDU refers to a message that requires protection against unauthorized manipulation or replay attacks such as messages with potential safety impacts. Such a message is considered as a normal message which would be sent without the usage of SecOC.

Secured PDU contains the Authentic PDU which needs to be protected and contains additional data to enable the receiving ECU to determine the integrity and authenticity of the original message. This is achieved by including the original message in the payload by adding the Authenticator (e.g. Message Authentication Code (MAC)) to the PDU. Optionally a header and the freshness value which was used to create the Authenticator can also be added to the PDU. The structure of such a PDU is illustrated in figure 4.15

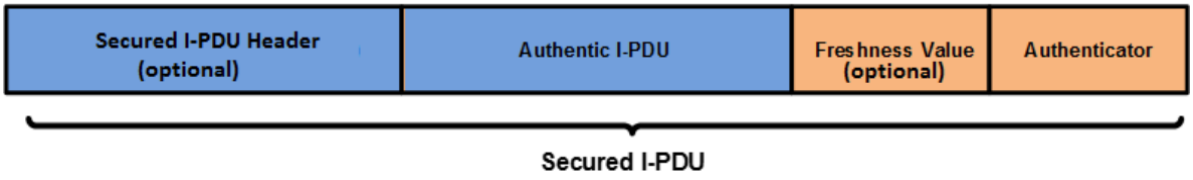

Figure 4.15.: Secured PDU structure [119]

Figure 4.16 shows the working mechanism of SecOC. The sender takes the PDU data and generates a MAC by using the counter (freshness) value and symmetric key K. The counter is then optionally added to the Secured PDU and the generated MAC is added as the Authenticator. [1] The PDU is then transmitted over the bus until the receiver extracts the Authentic PDU and optionally extracts the counter value from the PDU and compares it with the freshness value it received from the freshness manager. The receiver then calculates the MAC and accepts the PDU if it calculates the same result.

Considerations

In order to implement the SecOC mechanism, OEMs need to provide a centralized synchronization mechanism for all ECUs that use this method. Such a mechanism in turn needs its own protection mechanism.

[1] The MAC can also be added and transmitted in a truncated form. Hereby only parts of the complete MAC are generated which can be beneficial in resource restricted environments. Naturally, such an implementation is considered less secure than using at least 128 bit. [119]

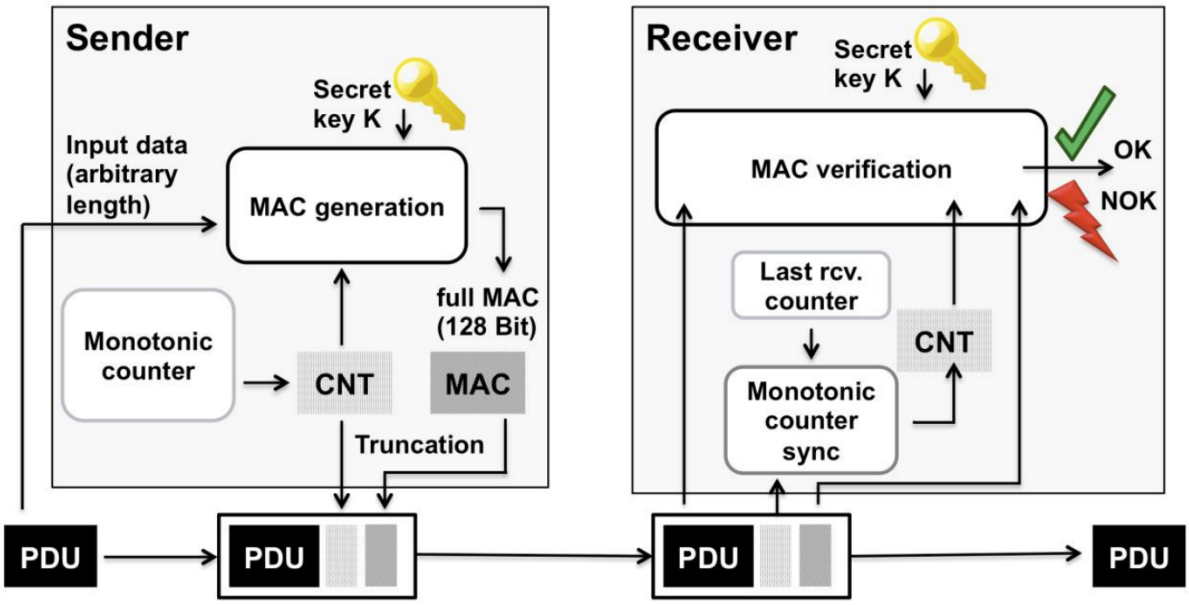

Figure 4.16.: SecOC mechanism [119]

Another challenge is the key management and distribution in such a system. Symmetric algorithms are well suited as they are more performant than asymmetric methods, but in turn leads to the compromisation of the entire system if the symmetric key of a single ECU is disclosed. One solution of this problem would be the distribution of individual keys for a logical group of messages as proposed by Quesnelle. [120]

Asymmetric keys can be used according to the AUTOSAR specification but signature generation/verification are considered as too slow to be practical. In this approach the MAC could be replaced by a signature which needs to be generated by the sender and verified by the receiver. As different algorithms for generation and verification are used, this would render truncation as impossible and therefore increase the message length.

In order to prevent the generation of too much traffic and verification efforts OEMs also need to thoroughly consider which PDUs should be protected.

4.10.6. Secured Boot

Security goals covered in this chapter: SG-30, SG-31

The goal of a secure boot mechanism is to preserve the reliability of embedded devices by ensuring the authenticity and integrity of the software during the boot process. By combining trusted hardware components 4.10.2 which can not easily be manipulated with cryptographic algorithms this mechanism can ensure that only validated software parts are executed on the system. [121] The HSM acts as the root

of trust or often refereed to as **trust anchor** for the secured boot functionality as it provides hardware support to withstand manipulations.

The secured boot mechanism is utilized in two distinct steps with different security challenges. In the first step, the **measuring** step is conducted which generates a comparison value (e.g. a hash value) based on the actual data of the SW within the memory. This value needs to be safely written into secured storage 4.10.2 to ensure the integrity of the calculated reference value. This needs to be considered when conducting a software update 4.10.9 because the measuring steps needs to be repeated as a change in the SW content will also lead to a different reference value which needs to be overwritten in the secured storage upon completion of the update. This also introduces new security challenges which need to be considered upon implementation.

During the second step (**verification**) which is conducted before the boot process, the hash function needs to calculate a reference value based on the currently flashed software which is then compared to the comparison value stored within the secure storage of the ECU. The software can be considered as authenticated, if the verification is successful. Otherwise it is considered as an authentication failure which needs to be treated by the chosen secured boot mechanism (e.g. by logging the authentication failure or stopping the boot process altogether). [121]

Authentic boot:

This mechanism checks the authenticity and integrity of software components during the boot process. In case a modification is detected, it does not employ direct countermeasures which hinder the boot process. It does however create a log entry which can be assessed by a third party including information about the origin of the error.

Secure boot:

This process also checks integrity and authenticity of the software during boot. The difference in this mechanism comes from the fact, that when detecting a modified software part or component, it actively executes preventive countermeasures like aborting the boot process or disabling certain functionalities. It therefore does not need a third party to evaluate the current state of the software and system.

By utilizing this mechanism, a **Chain of Trust** can be established. The HSM acts as the base root of trust which authenticates the boot code. The boot code can then furthermore authenticate the next boot stage (e.g. the bootloader of the ECU following the same concept. Each trusted boot stage therefore authenticates the next boot stage to ensure the integrity of the ECU software.

Another common goal of a secured boot process is to prevent software downgrades to older and potentially vulnerably software versions. Such a protection can be achieved by including meta data like

version numbers into the procedure. One solution would be a monotonic counter which is incremented for each version (e.g. a fuse that is burnt for each major version). [121]
Furthermore, the boot process shall verify that individual software parts are compatible to each other and all interdependencies between software entities are satisfied.

Considerations

Although the secured boot mechanism seems straight-forward, it also introduces new challenges and risks. It is not only challenging to manage different implementations of the mechanism of different manufacturers, but also increases the complexity regarding updatability of components. It also requires a secure root of trust in the form of a hardware protected security environment 4.10.2 which is not available in all devices.

4.10.7. Feature Activation/Deactivation

Security goals covered in this chapter: SG-11, SG-36

Feature Activation refers to the process of enabling vehicular features which are preinstalled but deactivated with the possibility of activating them securely with an activation code. Such a feature enables the activation of features on a pay-per-use model and also enables manufacturers to safe costs by using the same hard- and software in different vehicles with distinct features by enabling certain features only for specific models. Schramm and Wolf [122] propose a method to securely activate and deactivate such features by using cryptography to secure the activation codes. An example feature activation code and included data is depicted in figure 4.17. The security in the verification process comes from the usage of asymmetric cryptography in the form of certificates.
If the request to activate a certain feature is raised, the OEM can select a vehicle based on the Vehicle Identification Number (VIN) and define the feature which shall be activated based on a feature specific identifier. Its furthermore possible to add conditions to the feature activation like a timeframe of validity or maximum number of uses. To avoid replay attacks, a time stamp is added to the activation code. All this information is hashed and signed in the backend of the OEM and the resulting signature is added with all intermediate certificates required to verify the validity of the activation code. In the approach proposed by Schramm and Wolf [122], the OEM certificate authority holding the root certificate issues a supplier specific certificate which can be used to validate a feature activation certificate. By injecting the root certificate into the HSM of the ECU during production, the ECU is able to verify the certificate chain to verify the digital signature of the feature activation code.
Example use cases of such a feature include the activation of navigation features upon customer request

but feature activation could also be used as an additional layer of security for other features described in this chapter. Protected diagnostics 4.10.8 for example could utilize such a feature to permit or deny certain diagnosis requests depending on feature activation. Garages and repair shops could request the activation of enhanced diagnosis capabilities when necessary that should otherwise be restricted.

Purpose	Identifier	Description
Information fields	*VIN*	Vehicle Identification Number (typically up to 20 alphanumeric characters)
	Function ID	Identifier of the function that the code activates
	Conditions	Additional conditions of the activation code, such as its life cycle, maximum number of uses and so on
	Time Stamp	Time and date when the activation code was generated
	Generator ID	Identifier of the entity which has generated and signed the activation code
Protection fields	*Signature*	Message Authentication Code (MAC) or digital signature over all the data fields above for protection against any unauthorized manipulations.

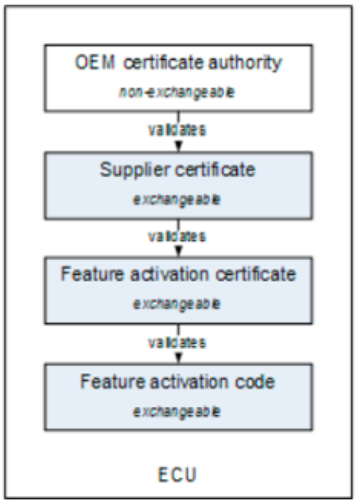

Figure 4.17.: Secure Feature Activation [122]

Considerations

In order to use secure feature activation, OEMs need to establish a backend capable of managing all individual features of their vehicle fleet and require a PKI 4.10.1. Additionally, as with all new features the possibility to exploit the feature activation mechanism also raises new security concerns and requires a hardware protected security environment 4.10.2 to provide security functionalities such as secure storage to store the necessary certificates to perform the signature verification of the feature activation codes.

4.10.8. Protected Diagnostics

Security goals covered in this chapter: SG-13

Diagnostic services 2.2.2 are used by external entities (also referred to as tester) to get information about the ECU and its operational status. The available services are specified in UDS which is also used to read and write data to the ECU and allows reprogramming. Therefore these services need to be protected against misuse by strong security mechanisms. ECUs must only accept diagnostic services from a trusted tester which requires authenticity even on remote access. [123] Vector proposes an approach where data and diagnostic services are restricted and locked before proper authentication and consecutively

unlocked if the tester can be authenticated by using certificates issued by a PKI 4.10.1. With such a mechanism it is also possible to limit the access permissions for individual testers (e.g. different services available to repair shops compared to manufacturer return analysis). Similar to the secure feature activation 4.10.7 use case, a chain of trust with digital certificates may be used to accomplish this. An example implementation from vector is depicted in figure 4.18.

The backend of the OEM needs to establish a secure connection to the Tester (e.g. the telematics unit of the vehicle) to initiate a diagnostic session. The tester sends a diagnostic authentication via standardized UDS (Service 29) and includes the diagnostic certificate which is signed by the certificate authority. This certificate can be verified by the ECU based on the root certificate stored in the HSM during production to obtain the public key. To verify the proof of ownership, the ECU sends a challenge to the tester which can only be solved by the possession of the corresponding private key. The tester therefore needs to forward the challenge to the backend where it is resolved. The proof of ownership is then transmitted to the tester which in turn forwards it to the ECU. The ECU can then verify the signature with the public key of the diagnostic certificate to obtain proof that it was signed by a higher level certificate authority. Depending on the additional information in the diagnostic certificate (e.g. roles, services to unlock), the ECU can provide access to a set of diagnostic services specified in the certificate.

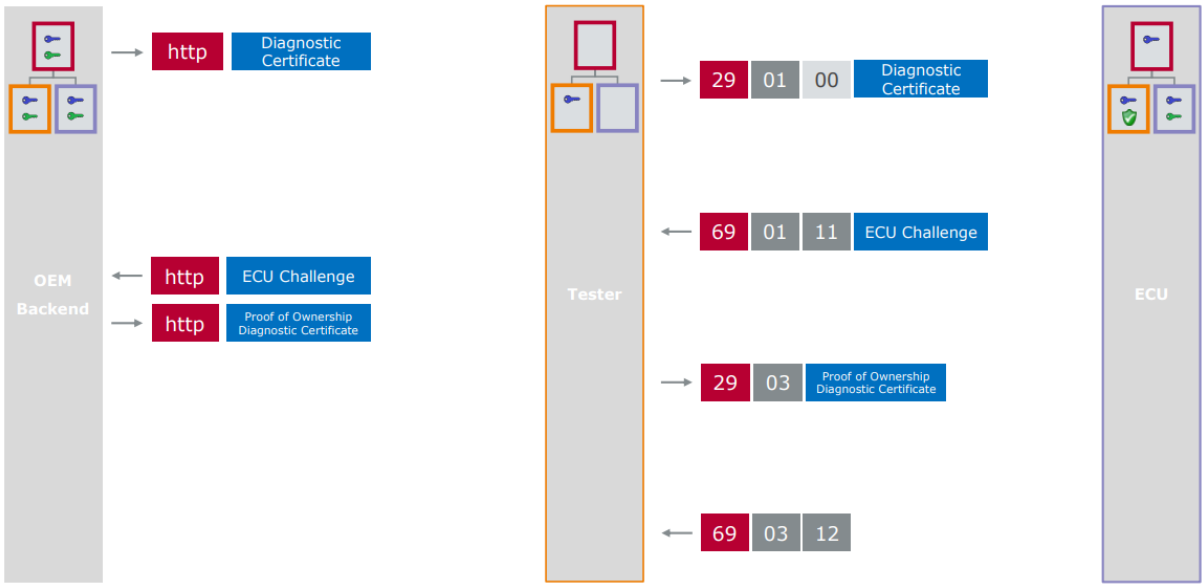

Figure 4.18.: Secure Diagnostics [123]

ISO 13400-2:2019 [124] specifices the requirements for secured diagnostic communication for clients and servers participating in Diagnostic communication over Internet Protocol (DoIP). DoIP edge nodes are responsible for the connection to the external entity over the internet by using the IP protocol. Such nodes communicate with DoIP gateways in the vehicle over IP which in turn transform the diagnosis

commands to the communication technology (e.g. CAN) within the vehicle sub-network.

Considerations

By enabling diagnostic functionalities without a physical connecting (e.g. plugging the tester into the OBD-II port) new security concerns like the security of the telematics unit and all other involved network components need to be considered.

4.10.9. Secure flashing

Security goals covered in this chapter: SG-30, SG-33

In order to provide updates to the firmware of ECUs, fix bugs or introduce feature improvements an update procedure needs to be in place that securely transfers and programs ECUs. The main objectives and requirements regarding such a procedure are described in UN Regulation No. 156 [88]:

The ECU shall ensure the integrity of any software update.

Only authenticated parties shall be able to alter ECU data.

A procedure to restore systems to their previous version in case of a failed flash attempt shall be implemented.

A solution to ensure the integrity of software content is proposed by Weimerskirch [125]. One assumed precondition in this procedure is the implementation of a PKI 4.10.1. The issuer of the software creates a hash of the program code and signs it with a secret key stored within the PKI or trust center of the OEM. The ECU has stored the matching public key within its HSM module which can be used to verify the signature of the software. Figure 4.19 provides a graphical illustration of this process.

After (1) software development a hash of the program code (2) is signed (3) by the secret key of the OEM. This package of program code and signature is then stored (4) in an appropriate structure before it can be downloaded (5) through various mechanisms before the signature is verified (6) with the public key stored in the ECU during production.

The program code may consist of multiple parts for different use cases and might contain multiple blocks with different attributes and signatures. The entire process is displayed in figure 4.20.

Before a programming session is initiated from an external system (tester, OTA update, ..) the external party needs to authenticate against the bootloader (e.g. by strong authentication mechanisms as described in 4.10.1). A certificate can be passed to the bootloader which is verified by the public key stored in the HSM. Each block or software part is then downloaded and decrypted, if necessary before they are

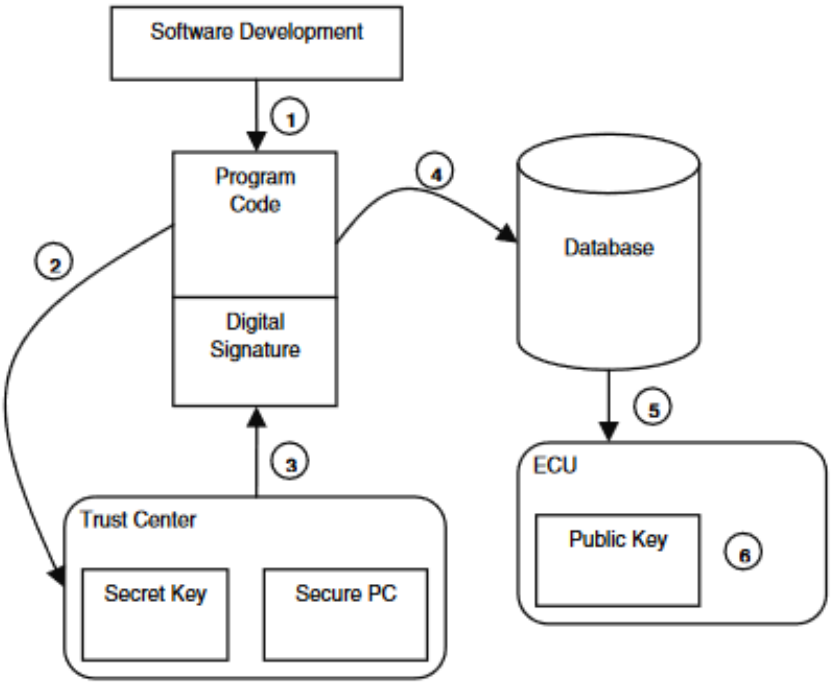

Figure 4.19.: Secure Software Download [125]

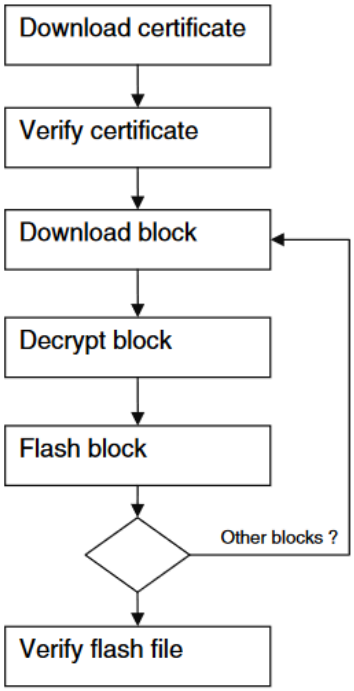

Figure 4.20.: Secure Flash Procedure [125]

flashed. After all blocks have passed, the ECU determines the hash value of the entire flash content and conducts a signature verification based on the signature provided by the external party. If the verification

is successful, the downloaded content is accepted and can be activated. Otherwise the bootloader needs to restore the flash to its previous state (e.g. by keeping a backup copy of the original code).

Considerations

The introduction of such a process needs to consider security at all levels. For example, misuses such as interrupting the flash procedure or modifying the software after successful signature verification need to be considered. A procedure to restore previous versions of software after a failed attempt also raises new security concerns like a modification of program data in the backup location.

4.10.10. OTA

Security goals covered in this chapter: SG-32

Over-the-Air updates 2.2.2 introduce the possibility to update vehicle components more frequently to fix functional as well as security issues and enable incident response to mitigate security vulnerabilities as they eliminate the need to drive the vehicle to a garage for a software update. Successful attacks, like a hack of the Tesla Model S [126] has demonstrated the need for a secure software update procedure for the entire automotive industry.

An OTA ecosystem includes backend and vehicle operations [55] whereas this paper mainly focuses on the operations occurring in the vehicle network, the general security principles which need to be followed are:

- All external and internal connections must be secure
- All involved entities need to be authenticated 4.10.1
- Integrity of the transmitted data needs to be ensured 4.10.5
- A verification of the installed firmware needs to be executed 4.10.9

Uptane is a secure software update system which is specifically designed for the automotive domain and considers resource restrictions of ECUs and addresses a comprehensive threat model. [127] The proposed software update workflow is described by referring to figure 4.21 [2] which provides an overview of an example architecture. Whilst the vehicle is running, the primary ECU (e.g. telematics unit) establishes a secure connection to the backend and downloads the latest relevant software update metadata. It checks the metadata of the software updates like release version, timestamp, etc and furthermore downloads the

[2] Partial verification refers to a less secure method of verifying an update in case of limited resources or components

actual images including the firmware which shall be installed on different ECUs. It verifies the content of these images by verifying the signature of the downloaded data by the utilization of a certificate based chain of trust. It then informs the secondary ECUs about available software updates. Upon stopping or restarting the vehicle, the driver is informed about a pending update and may choose to install it whenever suitable. Now that the software content is available to the ECUs the secure flashing procedure as described in 4.10.9 needs to be adhered to in order to securely update the firmware.

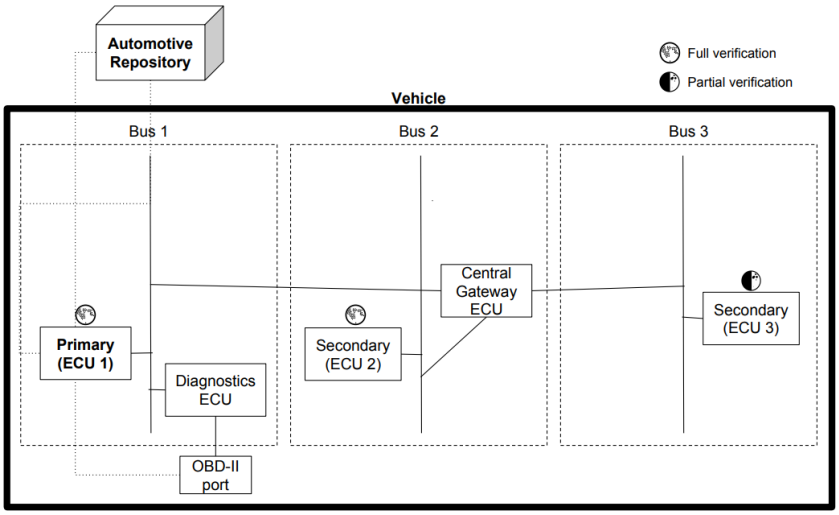

Figure 4.21.: OTA systematic architecture [127]

Considerations

Introducing an OTA update process provides various benefits as explained in 2.2.2 but also raises new security challenges. The support of such a system needs to be securely implemented by multiple ECU suppliers and requires protected communication channels to the backend. Additionally, the management and provisioning of software updates from various manufacturers and secure signature generation for all images is posing a challenge. The requirements towards the primary ECU regarding enough storage space for all available software images also increases cost of such components.

4.10.11. Secure Logging

Security goals covered in this chapter: SG-18, SG-29

ISO/SAE 21434 defines cyber security incident response activities in the operations and maintenance chapter. In order to enable this functionality, security critical events such as failed authentication attempts or manipulated flashing of software need to be recorded in a secure environment. The secure memory area or secure storage provided by he HSM 4.10.2 is ideally suited to record such events in a

confidential and integrity protected environment. To achieve such a functionality, the HSM needs to implement a feature to generate log entries itself but also needs to implement interfaces for the application software to report important security events external from the security module.

The security module also needs to implement a functionality to provide this information to authenticated participants by encrypting the data within the secured storage to ensure confidentiality. This can be achieved by the implementation of a diagnosis feature, only accessible under certain circumstances further outlined in section 4.10.7.

4.10.12. Hardware attestation

Security goals covered in this chapter: SG-28

Hardware attestation refers to a defense mechanism against software attacks in cases where for example the firmware of a safety critical ECU is manipulated. Such a system is typically realized as a challenge-response mechanism between a verifier and prover. Kohnhauser et al. [4] propose a solution to ensure the safe and secure operation of ECUs by an attestation scheme which utilizes a trusted master ECU which verifies the software integrity of all safety-critical ECUs in the vehicle upon every start of the vehicle and consecutively only starts the engine in case of a successful verification. Figure 4.22 shows how the attestation scheme works for both simple (reduced security feature set) and advanced ECUs (with a Hardware protected security environment 4.10.2).

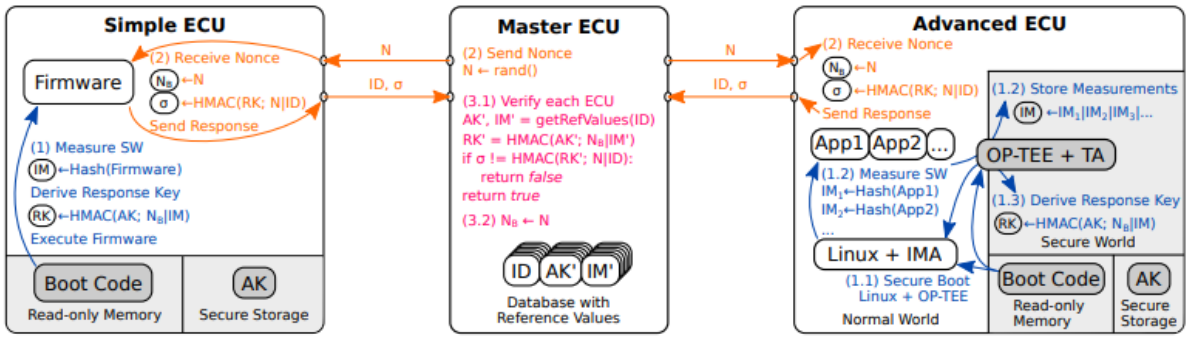

Figure 4.22.: ECU Attestation [4]

The master ECU is responsible for verifying the software integrity of all ECUs which participate in the protocol in a three step process:

Likewise to the secured boot process 4.10.6 each ECU computes a hash value of the currently installed firmware and stores it in IM (integrity measurement variable). The ECU then computes a Hash-based Message Authentication Code (HMAC) over the current nonce (random number from the master ECU) and IM by using the attestation key AK which needs to be stored in secured storage and is only known

to the ECU and the master to generate the response key RK.
In the second step, the master ECU generates a new nonce N as a challenge and broadcasts it to all participating ECUs which is a preventive measure against replay attacks. Upon receiving the new challenge, the ECU overwrites the current nonce which causes the resulting response key RK to update. The ECU then generates the attestation response which consists of a HMAC computed over the ECU specific ID, the old response key RK and and updated nonce which is then sent as sigma to the master.
The master then verifies the responses upon reception by extracting the attestation key AK and previous integrity measurement varialbe IM of each participant from its secure storage. Based on the received ID, it retrieves IM, AK and the previous nonce NB to compute the expected response key RK itself. It then calculates sigma with these values and compares it to sigma received from the ECU. If the values match, the attestation of the ECU can be considered as valid.

Considerations

Sensors and actuators often do not provide the required security environment to facilitate such a mechanism. Additionally, the master ECU is a central point of failure in this approach and needs to be protected by strong security measures. The system is also prone to denial of service attacks as a failed attestation would hinder the engine from being started. Furthermore, the system adds additional complexity in a software update procedure as the hash value of the IM would change in such a scenario which introduces the necessity of communicating this change to the master ECU.

4.11. Risk re-evaluation

To evaluate the effectiveness of the defined countermeasures 4.10, the risk assessment is repeated to demonstrate that the introduction of security measures lowers the risk of the identified threat scenarios. The evaluation result shown in table 4.17 therefore considers the implementation of the identified security measures during the feasibility assessment. Annex B includes details about the result of the attack step and threat scenario feasibility assessment.

ID	Threat scenario	Attack feasibility assessment			Impact Rating			Risk value
		Value	Rating		Scenario	Rating		
T1	Readout of HSM software	57	Very Low	0	P, F	Major	1,5	1
T2	Readout of sensor data	8	High	2	P	Negligible	0	1
T3	Readout of message data	57	Very Low	0	P, F	Major	1,5	1
T4	Readout of ECU software	57	Very Low	0	P, F	Major	1,5	1
T5	Readout of configuration data	57	Very Low	0	P	Negligible	0	1
T6	Modification of HSM software	57	Very Low	0	S	Severe	2	1
T7	Modification of HSM content	57	Very Low	0	S	Severe	2	1
T8	Disclosure of crypto material	57	Very Low	0	S	Severe	2	1
T9	Modification of crypto material	57	Very Low	0	S	Severe	2	1
T10	Modification of sensor data	34	Very Low	0	S	Severe	2	1
T11	Modification of message data	57	Very Low	0	S	Severe	2	1
T12	Modification of ECU software	57	Very Low	0	S	Severe	2	1
T13	Modification of ECU functionality	34	Very Low	0	S, F	Severe	2	1
T14	Modification of configuration data	57	Very Low	0	S	Severe	2	1
T15	DoS on sensor data	17	Medium	1,5	O	Moderate	1	2,5
T16	DoS on message data	17	Medium	1,5	O	Moderate	1	2,5
T17	DoS on ECU functionality	17	Medium	1,5	O	Moderate	1	2,5
T18	Readout of HSM content	57	Very Low	0	O	Negligible	0	1

Table 4.17.: Risk re-evaluation result

The threat scenarios T15 - T17 still raise concerns as the threat and risk evaluation results in an intermediate risk value of 2,5. Palanca et al. [128] discuss the issues leading to this risk in detail in their paper. The risk in these threat scenarios originates from the design of the CAN bus 2.1.5 which allows an attacker to execute a DoS attack without the need of transmitting a complete data frame. The paper discusses this vulnerability in further detail and proposes an experimental proof-of-concept, but concludes that no reasonable countermeasure for this attack currently exists.

5. Conclusion

This paper presents and analyzes emerging vehicle technologies and how their functionality impacts ECUs in vehicles. The ISO/SAE 21434 "Road vehicles - Cybersecurity engineering" standard [13] published in mid 2021 is utilized to determine security threats for individual ECUs which originate from these emerging vehicle technologies based on a threat and risk assessment methodology using attack trees proposed in the standard. The threats are furthermore used to derive security goals/requirements for ECUs to elicitate countermeasures for the identified threat scenarios. The effectiveness of the proposed countermeasures is then validated by repeating the risk assessment considering the implementation of the security measures.

5.1. Discussion

A literature analysis of current standards and recommendations has shown that whilst various documents regarding automotive security have been published, only very few define specific requirements or technical measures for individual components and require the OEM to implement a suitable solution based on generic process descriptions. By using the approach followed in this research it was possible to determine specific technical countermeasures for security threats concerning ECUs. The results of this work show that multiple distinct security functionalities are required to provide an adequate security level for ECUs. It becomes evident, that the use of a hardware protected security environment 4.10.2 is essential to provide a secured environment as a root of trust for all security functionalities. Furthermore, this research illustrates the complexity of securing vehicular functionalities against threats arising from emerging vehicle technologies as all identified countermeasures only consider the first of five layers in the layered security approach 2.3.2. The identified countermeasures 4.10 also show that strong interdependencies exist between them which need to be considered and addressed during development and implementation. The asset identification 4.2 regarding emerging vehicle technologies and their security impact towards ECUs has revealed, that the impacted assets do not differentiate from traditional vehicular technologies. The identified technologies require the same protection mechanisms as traditional features but increase the attack surface due to increased communication to external networks. Due to this fact, OEMs need to

pay special attention to the telematics unit, gateways and network security.
OEMs and manufacturers also need to consider the security threats which arise from the implementation of new security measures and also face issues that are unique to the automotive landscape. Constraints of embedded devices such as resource restrictions and real-time requirements make the implementation of security measures more difficult than in traditional IT environments. At the same time both hardware and development costs of the installed components are significantly increased by introducing security measures to all components. Considering compatibility and cost, adopting all security measures against automobile hacking seems unrealistic [129]. The added maintenance effort required to securely manage a fleet of vehicles also needs to be funded by appropriate business models like maintenance contracts. [94]
The distributed development of hard- and software between OEMs Tier-1, Tier-2 and Tier-3 suppliers also increases the complexity of achieving an adequate security level across the entire vehicle lifecycle.
In conclusion, the security measures identified in this work can be considered as the foundation to ensure safe vehicle operation as they protect the integrity and intended functionality of ECUs in modern vehicles. However, these measures only secure the infrastructure on which further security mechanisms need to be established to achieve an appropriate security level. Considering the increasing complexity of vehicle functionalities and unique challenges in the automotive landscape, OEMs face many challenges towards securing vehicles of the future.

5.2. Limitations

Whilst the methodology used in this paper was able do identify and define security measures for individual ECUs, the work done in this research does not cover all security relevant aspects in the automotive landscape. In order to produce technical specification recommendations, the taken approach required the definition of a narrow scope which results in the exclusion of important security relevant components like the in-vehicle network, gateways and the telematics control unit. Furthermore, the used methodology can not guarantee that all technologies and threats towards ECUs are considered in the threat and risk analysis and therefore may result in an incomplete countermeasure definition.
ISO 21434 2.4.2 defines very detailed parameters regarding the classification of attack feasibility in the attack potential-based approach, but the rationale applied during the feasibility rating partly relies on subjective opinions based on experience in the field.
Time and resource restrictions also limit the possibility for the evaluation of countermeasure effectiveness to a theoretical assessment. The effectiveness of the defined countermeasures can only be evaluated on a technical level upon implementation and tested during operation.

5.3. Future work

The focus of this paper was to identify security measures for individual ECUs. However, such devices are only part of the entire vehicle and its environment and surrounding infrastructure.
The vehicular network including domain controllers, gateways and the telematics unit are especially important considering the increasing need for communication towards external networks and provide numerous topics for further research.
Some proposed countermeasures require a central processing unit for various functionalities with special security requirements like the primary ECU responsible for storing and distributing images in OTA update use cases. A fallback mechanism for OTA updates and secure flashing procedures is also an interesting topic for further research as a failed flash attempt would result in an inoperable vehicle if devices do not provide the storage capacity required for the current firmware and the updated version. Furthermore, as mentioned in the considerations 4.10.9, such a fallback mechanism needs to be protected against manipulation as well to prevent attackers from modifying the backup firmware.
As mentioned in the work of Palanca et al. [128] the CAN network, even though proven in use, has design flaws which need to be addressed in order to prevent DoS attacks.
Securing the communication and network infrastructure outside of the vehicle also needs to be addressed and standardized as autonomous systems need to be able to rely on the data provided by RSUs.

To summarize, the automotive landscape is facing security challenges within embedded devices, software, inter- and intravehicle networks, global networks, backend systems, network infrastructure and autonomous driving. In addition, life-threatening situations can result from misbehaviour of the system underlining the need for more research in this field to develop safe and secure vehicles in the future.

A. Threat scenario attack paths

This addendum includes all relevant attack paths identified during the threat and risk analysis in section 4.5.

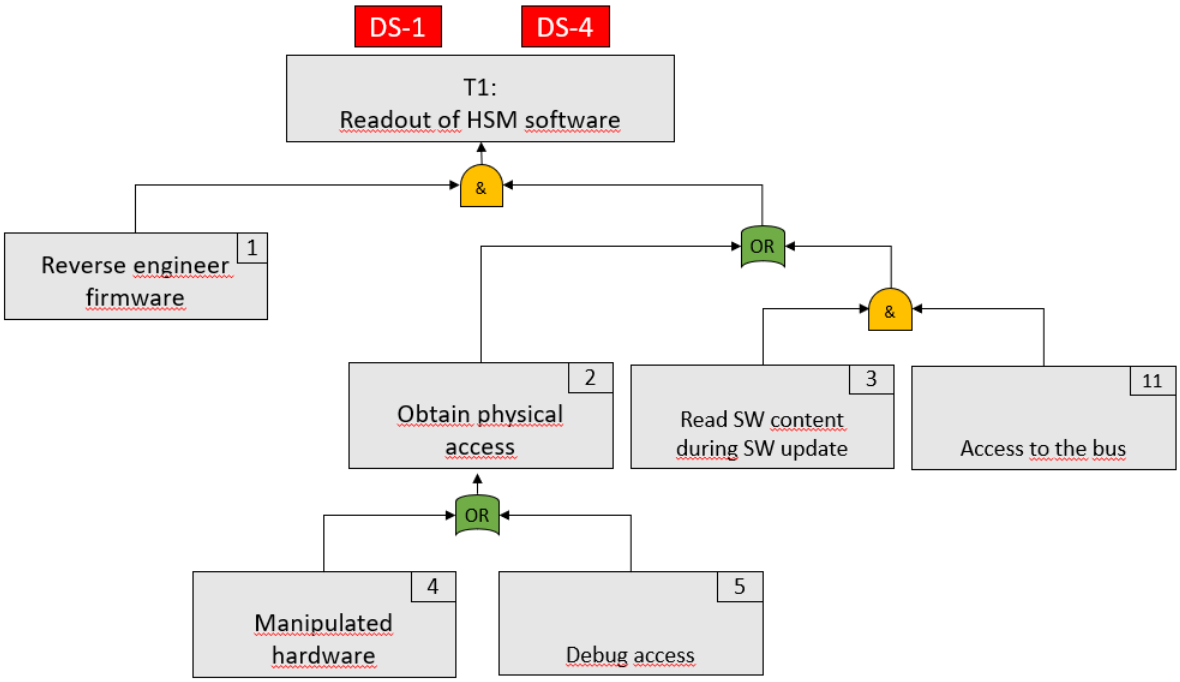

Figure A.1.: Threat scenario 1

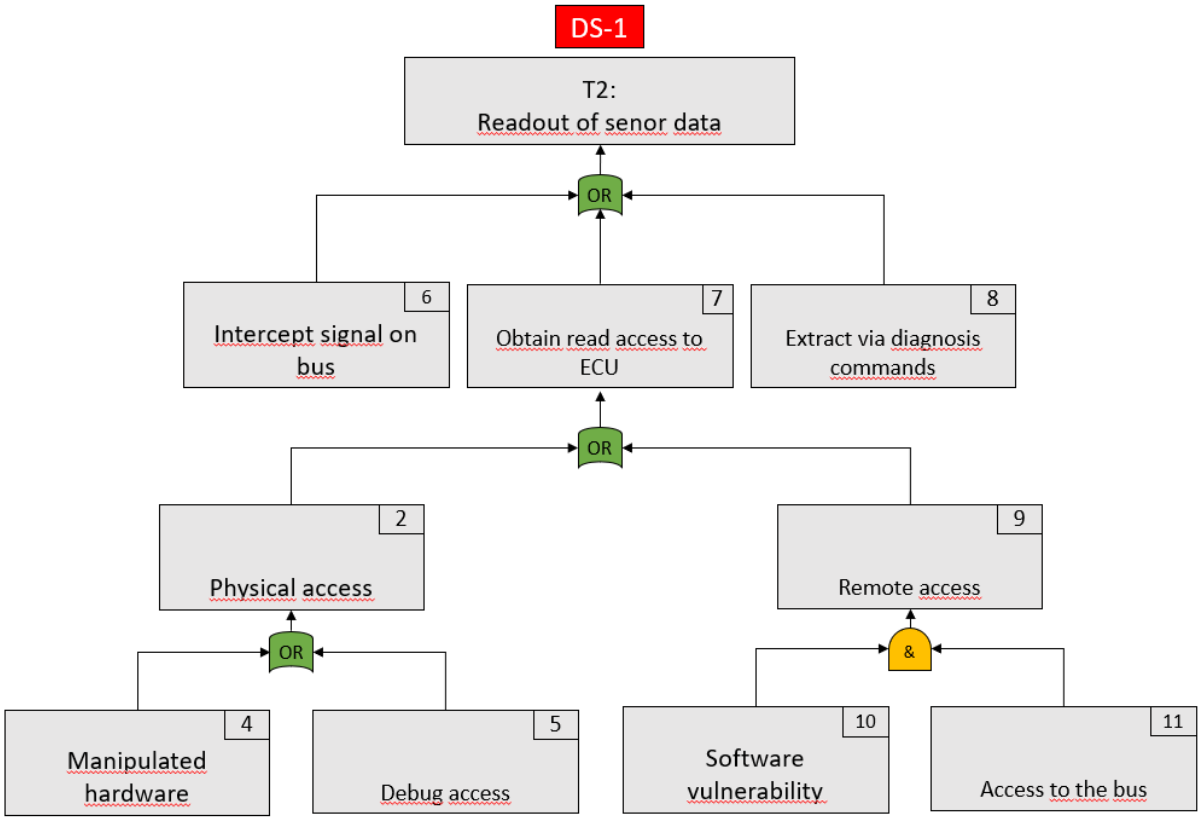

Figure A.2.: Threat scenario 2

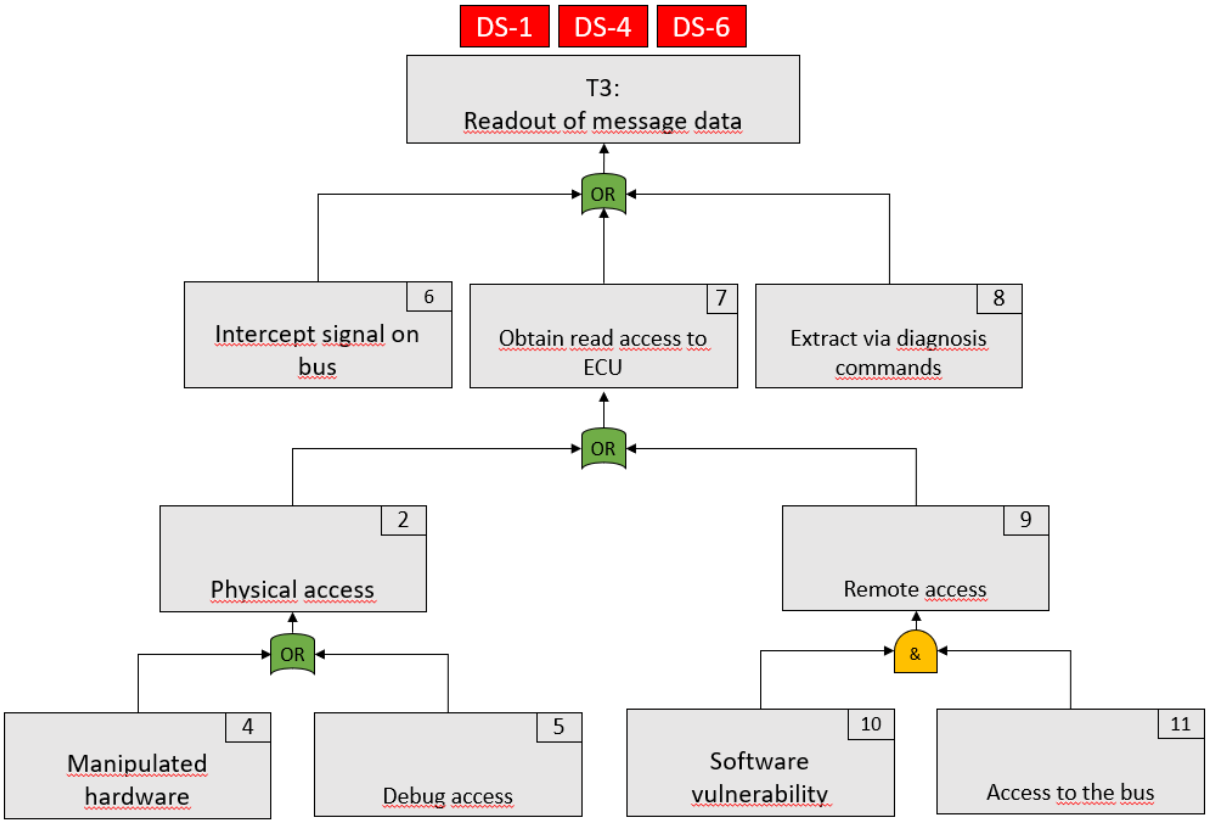

Figure A.3.: Threat scenario 3

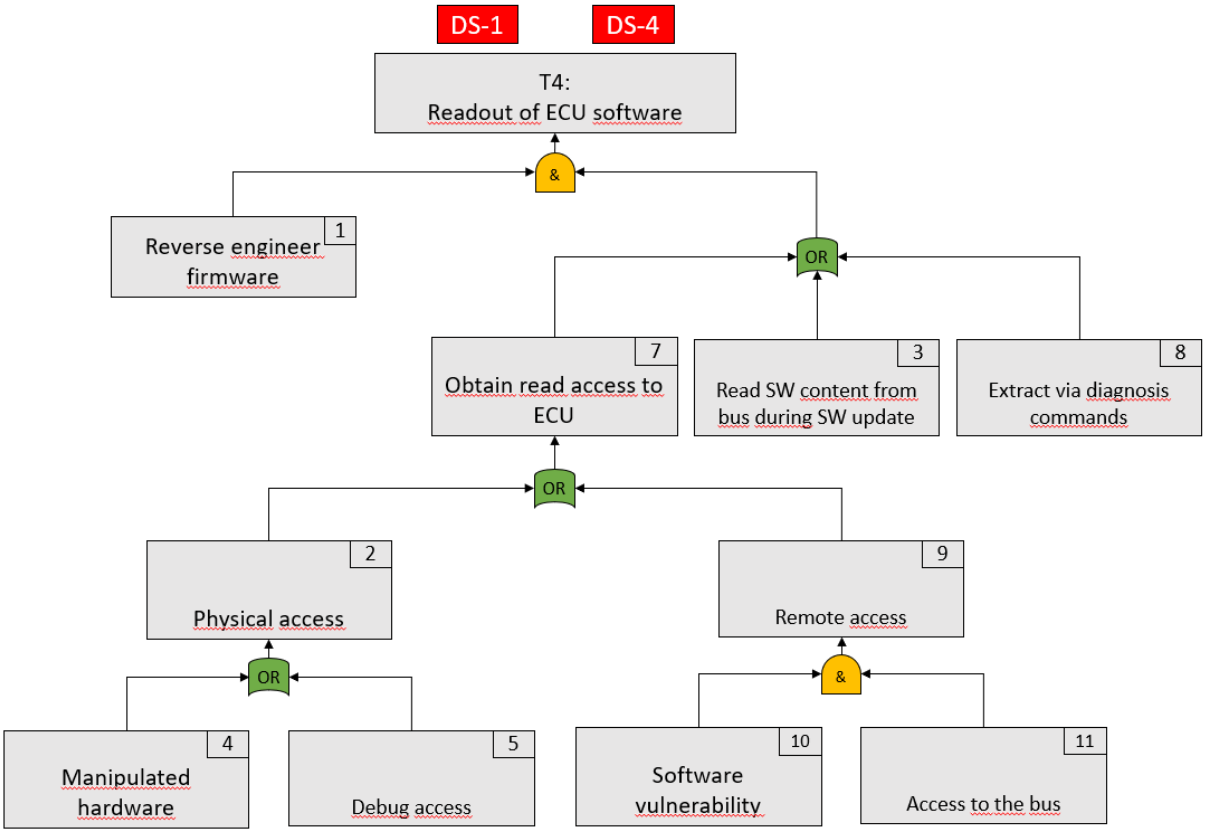

Figure A.4.: Threat scenario 4

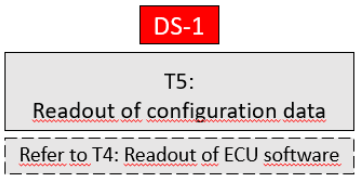

Figure A.5.: Threat scenario 5

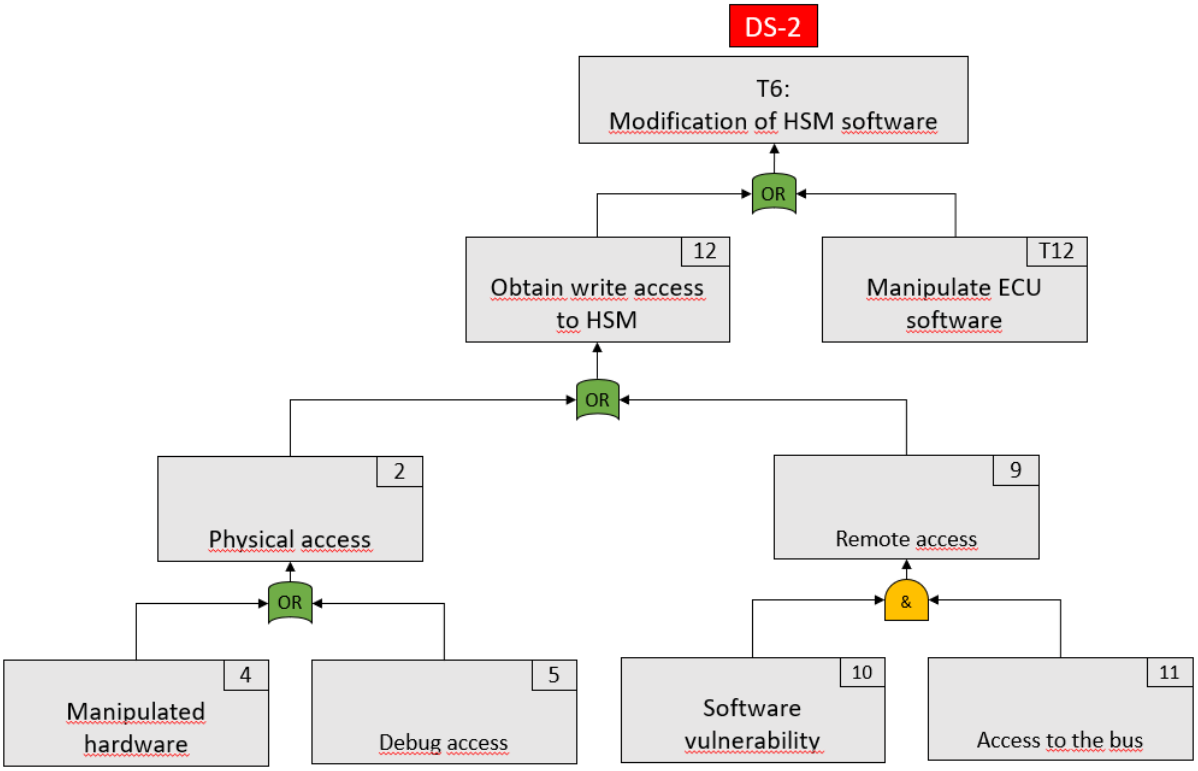

Figure A.6.: Threat scenario 6

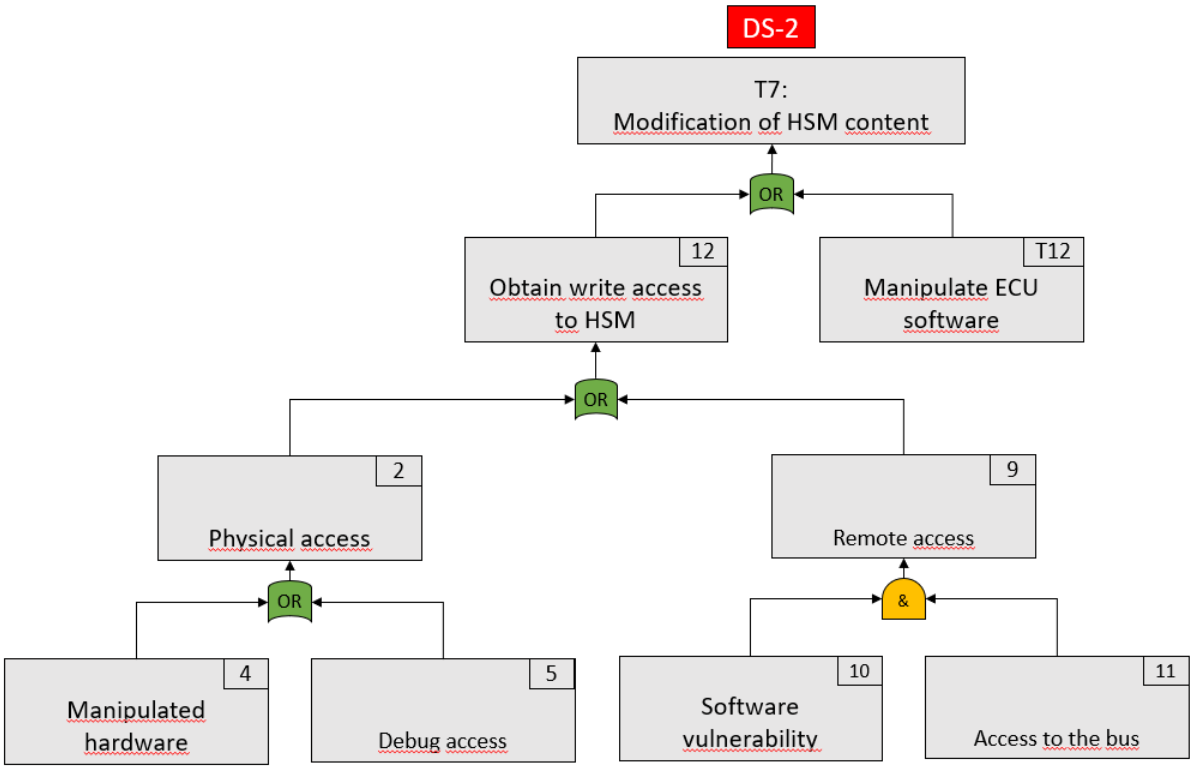

Figure A.7.: Threat scenario 7

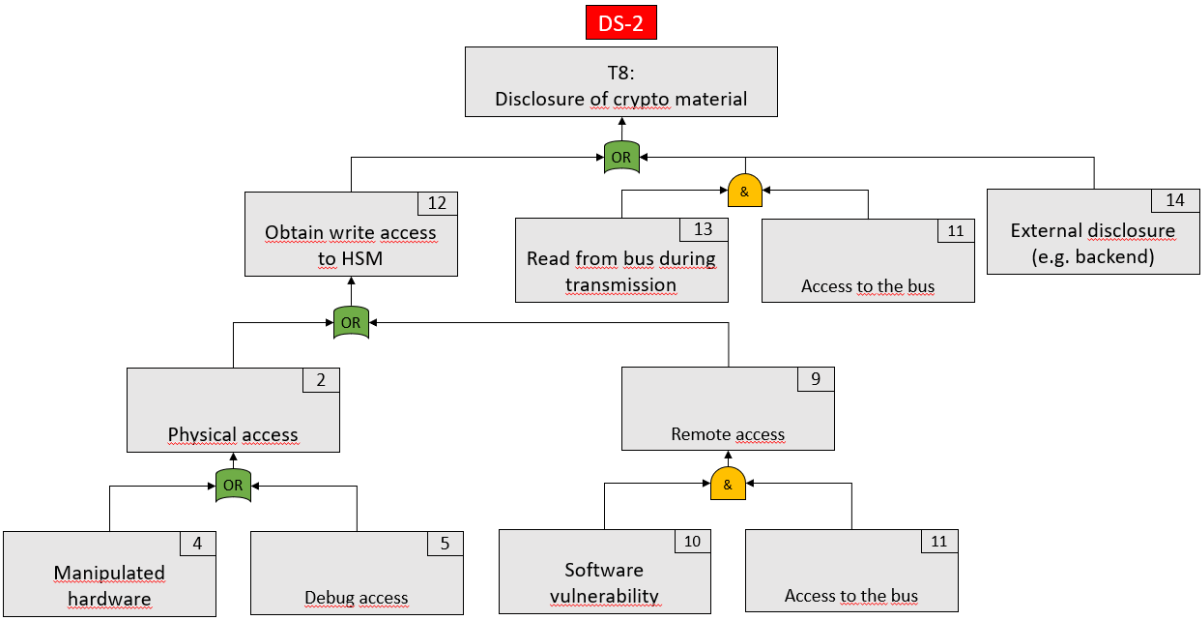

Figure A.8.: Threat scenario 8

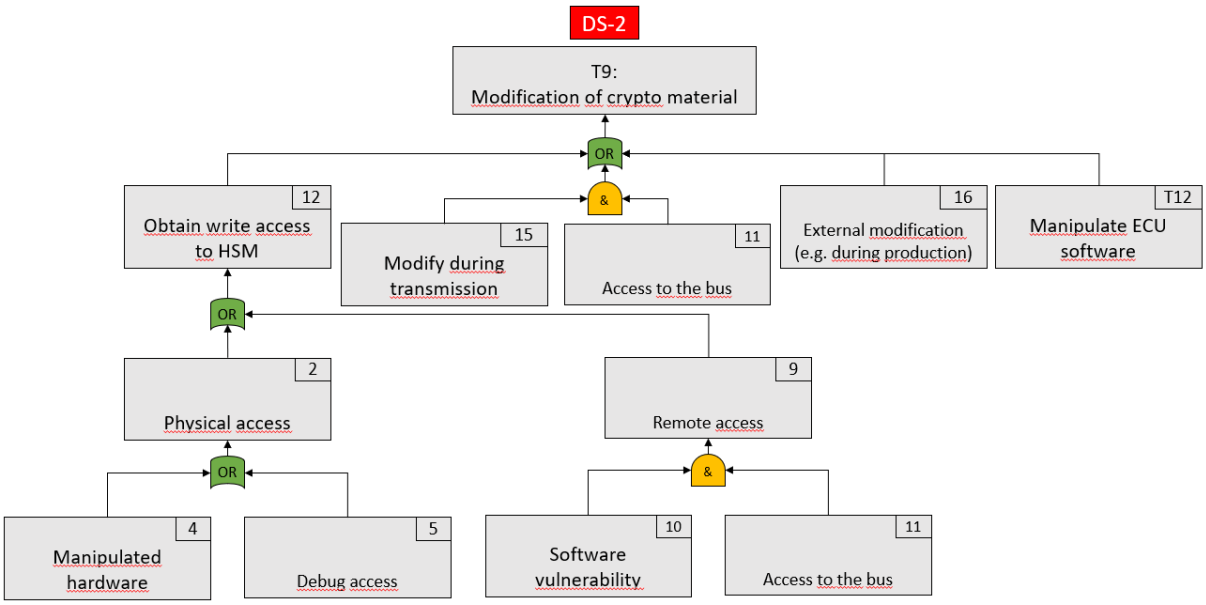

Figure A.9.: Threat scenario 9

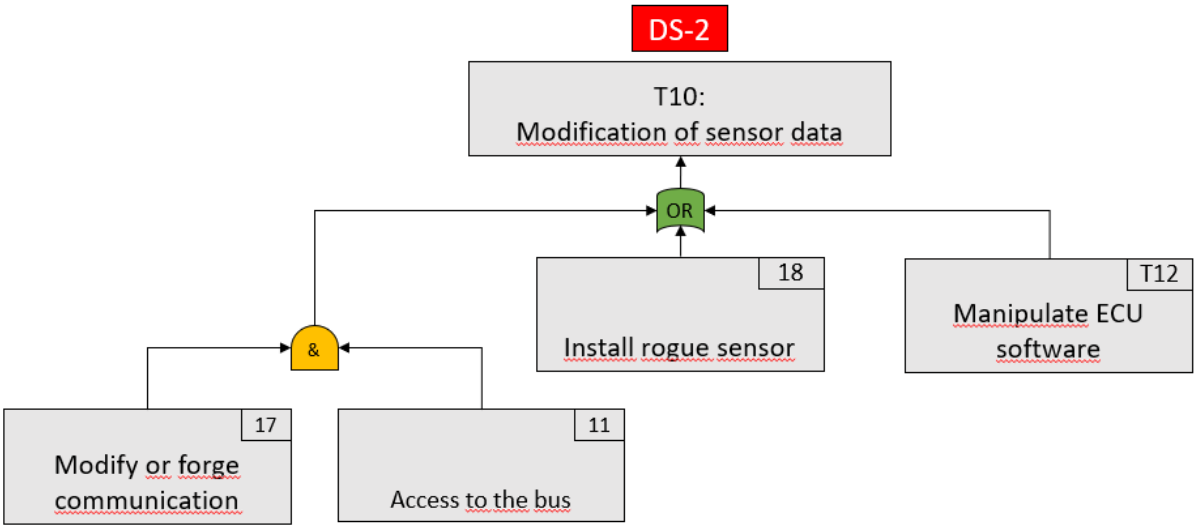

Figure A.10.: Threat scenario 10

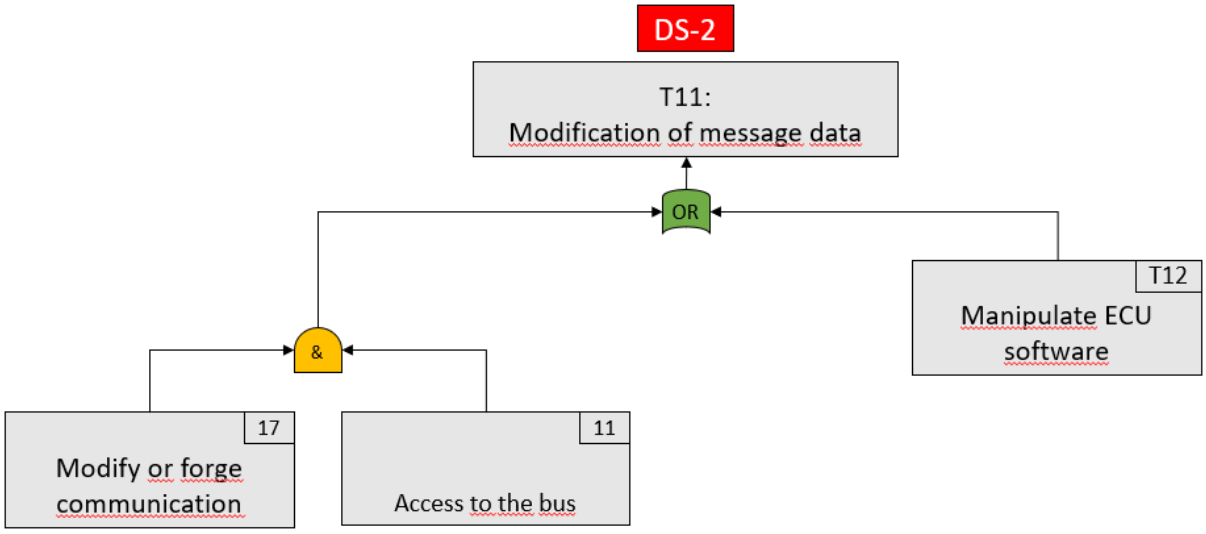

Figure A.11.: Threat scenario 11

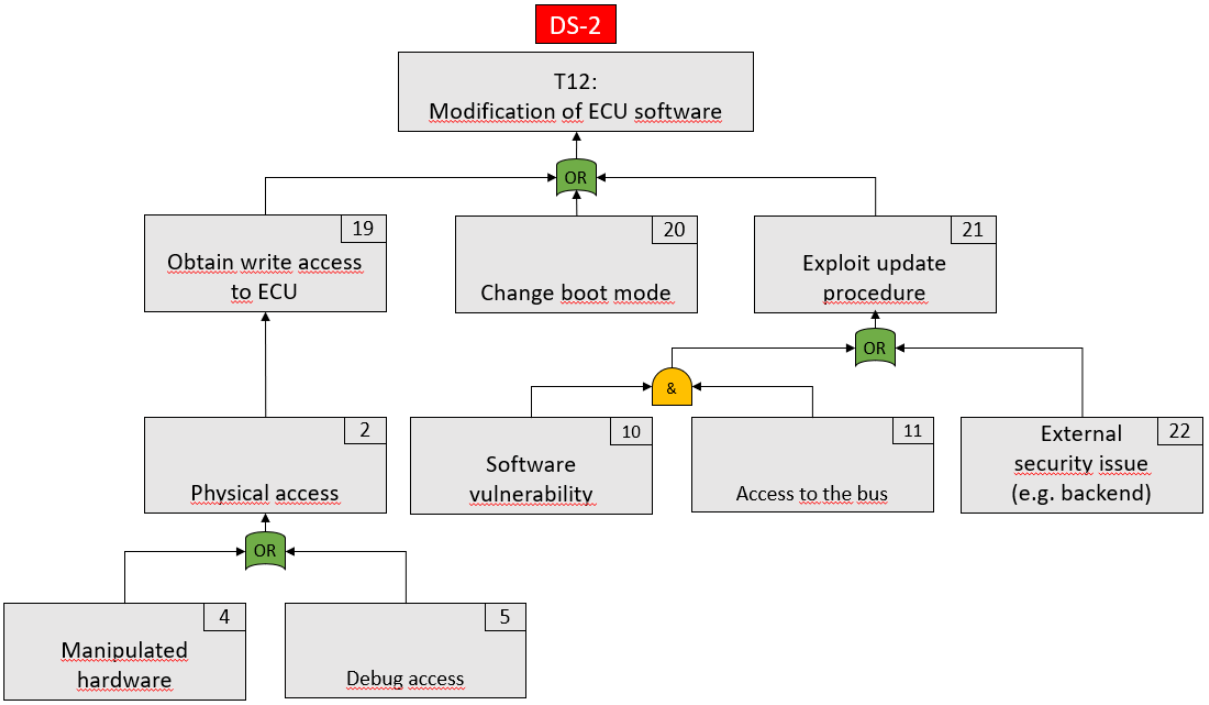

Figure A.12.: Threat scenario 12

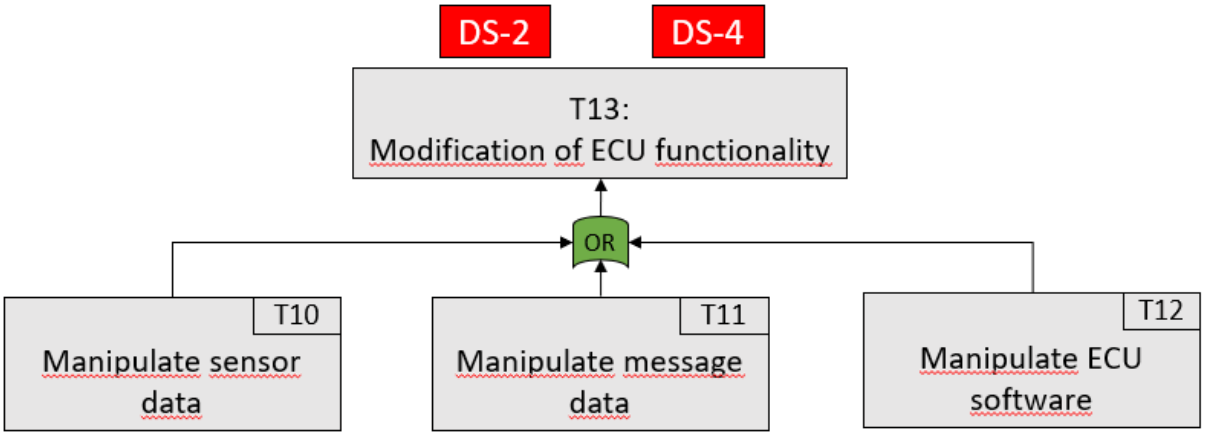

Figure A.13.: Threat scenario 13

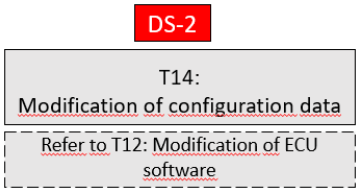

Figure A.14.: Threat scenario 14

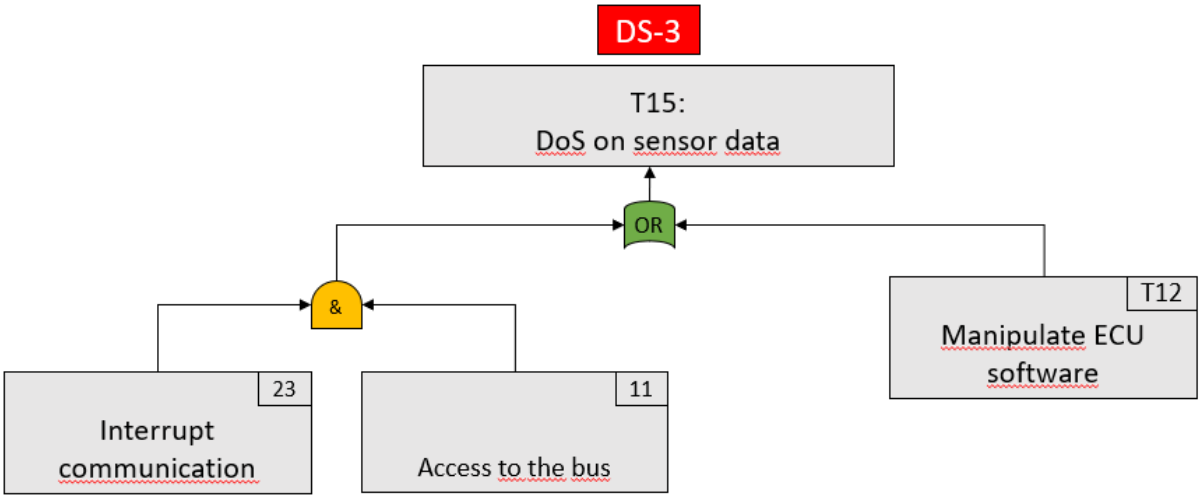

Figure A.15.: Threat scenario 15

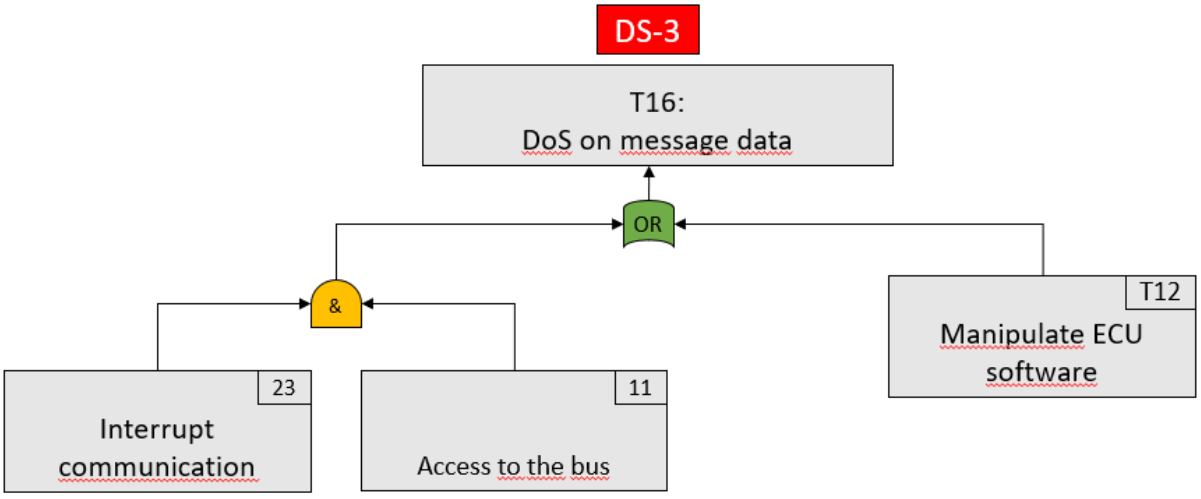

Figure A.16.: Threat scenario 16

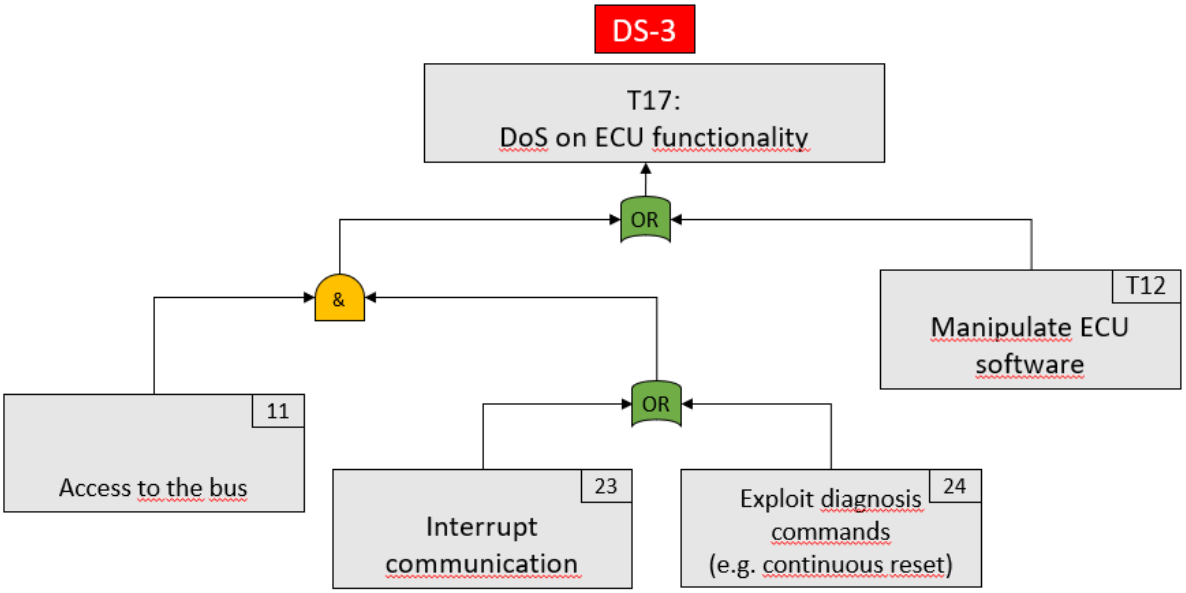

Figure A.17.: Threat scenario 17

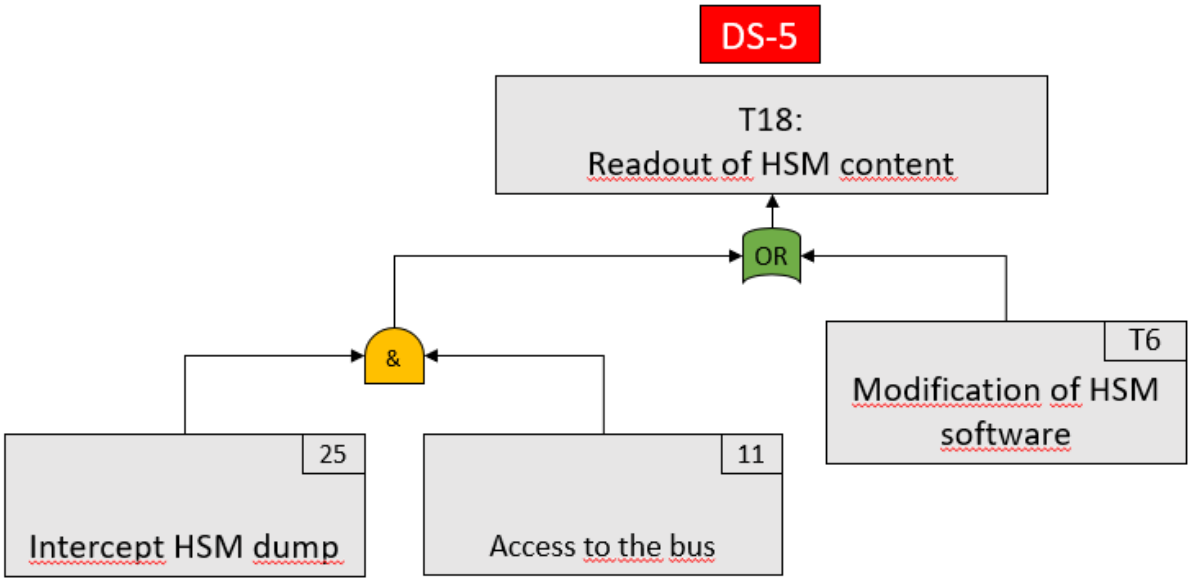

Figure A.18.: Threat scenario 18

B. Threat re-evaluation

This addendum includes the intermediate steps of the threat re-evaluation 4.11 process. Table B.1 shows the attack feasibility assessment of the attack steps outlined in Annex A whereas table B.2 provides an overview of the determined attack feasibility for each threat scenario 4.4.

ID	Attack step	Attack feasibility assessment					
		Time	Expertise	Knowledge	Access	Equipment	Aggregate
1	Reverse engineer firmware	4	6	3	0	0	13
2	Obtain physical access	19	8	11	10	9	57
3	Read SW content during SW update	19	8	11	10	9	57
4	Manipulated hardware	19	8	11	10	9	57
5	Debug access	19	8	11	10	9	57
6	Intercept signal on bus	0	3	0	1	4	8
7	Obtain read access to ECU	19	8	11	10	9	57
8	Extract via diagnosis commands	19	8	11	10	9	57
9	Remote access	19	8	11	10	9	57
10	Software vulnerability	19	8	11	10	9	57
11	Access to the bus	0	0	0	1	0	1
12	Obtain write access to HSM	19	8	11	10	9	57
13	Read from bus during transmission	19	8	11	10	9	57
14	External disclosure (e.g. backend)	19	8	11	10	9	57
15	Modify during transmission	19	8	11	10	9	57
16	External modification (e.g. during production)	19	8	11	10	9	57
17	Modify or forge communication	19	8	11	10	9	57
18	Install rogue sensor	1	8	11	10	4	34
19	Obtain write access to ECU	19	8	11	10	9	57
20	Change boot mode	19	8	11	10	9	57
21	Exploit update procedure	19	8	11	10	9	57
22	External security issue (e.g. backend)	19	8	11	10	9	57
23	Interrupt communication	0	6	3	4	4	17
24	Exploit diagnosis commands (e.g. continuous reset)	19	8	11	4	4	46
25	Intercept HSM dump	19	8	11	10	9	57

Table B.1.: Attack step feasibility assessment considering countermeasures

ID	Threat scenario	Attack feasibility assessment					
		Time	Expertise	Knowledge	Access	Equipment	Aggregate
T1	Readout of HSM software	19	8	11	10	9	57
T2	Readout of sensor data	0	3	0	1	4	8
T3	Readout of message data	19	8	11	10	9	57
T4	Readout of ECU software	19	8	11	10	9	57
T5	Readout of configuration data	19	8	11	10	9	57
T6	Modification of HSM software	19	8	11	10	9	57
T7	Modification of HSM content	19	8	11	10	9	57
T8	Disclosure of crypto material	19	8	11	10	9	57
T9	Modification of crypto material	19	8	11	10	9	57
T10	Modification of sensor data	1	8	11	10	4	34
T11	Modification of message data	19	8	11	10	9	57
T12	Modification of ECU software	19	8	11	10	9	57
T13	Modification of ECU functionality	1	8	11	10	4	34
T14	Modification of configuration data	19	8	11	10	9	57
T15	DoS on sensor data	0	6	3	4	4	17
T16	DoS on message data	0	6	3	4	4	17
T17	DoS on ECU functionality	0	6	3	4	4	17
T18	Readout of HSM content	19	8	11	10	9	57

Table B.2.: Threat scenario feasibility assessment considering countermeasures

C. Glossary

ADAS	Advanced Driver Assistance Systems. v, 11, 19, 47, 48
AUTOSAR	Automotive Open System Architecture. v, 9, 10
BSW	Basic Software. 47
CAN	Controller Area Network. v, 6
CSR	Certificate Signing Request. 74, 75
DoIP	Diagnostic communication over Internet Protocol. 85
DoS	Denial of Service. 92, 95
DSRC	Dedicated Short-Range Communication. 16
ECU	electronic control unit. iii–v, 1, 3–5, 20, 21, 26–29, 37–39, 42–46, 48, 49, 51, 59, 69, 70, 74–91, 93–95, 111
FOTA	Firmware Over The Air. 20
HMAC	Hash-based Message Authentication Code. 90, 91
HSM	Hardware Security Module. v, 5, 47, 71, 74–76, 78, 81–83, 85, 86, 89, 90, 111
IDS	Intrucsion Detection System. vi, 76–78
IoT	Internet of Things. 22
ISO	International Standardization Organization. 6
ITS	Intelligent transport systems. 15, 19, 48, 52, 53

LDM	Local Dynamic Map. 54
LIDAR	Light Detection and Ranging. 14
LIN	Local Interconnect Network. v, 6
MAC	Message Authentication Code. 80, 81
MOST	Media Oriented System Transport. v, 7
NHTSA	National Highway safety Transportation Agency. 14
NvM	Non-volatile Memory. 47
OEM	original equipment manufacturer. 2, 3, 9, 20, 21, 27, 28, 75, 80, 81, 83–85, 93, 94
OTA	Over the Air. 19, 20, 86, 88, 89, 95
PASTA	Process for Attack Simulation and Threat Analysis. 31
PDU	Protocol Data Unit. 79–81
PFlash	Program Flash. 46, 47
PKI	Public Key Infrastructure. 73, 75, 79, 84–86
RADAR	Radio Detection and Ranging. 12
RAM	Random Access Memory. 47
RFID	Radio frequency identification. 18
RSU	Roadside Unit. 19, 95
SaaS	Software-as-a-Service. 20
SecOC	Secure Onboard Communication. vii, 79–81, 111
SIEM	Security Incident and Event Management. 77
SOC	Security Operation Center. 77
SOP	Start-of-Production. 20
TARA	Threat and Risk Analysis. v, 30, 39, 40, 45
TDMA	Time Division Multiple Access. 8
ToE	Target of Evaluation. 42, 47, 48, 51–54, 70, 72

TRNG	True Random Number Generator. 72
UDS	Unified diagnostic services. 21, 84, 85
V2I	Vehicle-to-Infrastructure. 17, 19
V2P	Vehicle-to-Pedestrian. 17, 18
V2V	Vehicle-to-Vehicle. 17, 19
V2X	Vehicle-to-Everything. 17, 19
VANET	Vehicular Ad hoc network. 16–18, 26
VIN	Vehicle Identification Number. 83

List of Figures

List of Tables

Bibliography

[1] C. Paunov and S. Planes-Satorra, "How are digital technologies changing innovation?" no. 74, 2019. [Online]. Available: https://www.oecd-ilibrary.org/content/paper/67bbcafe-en

[2] M. Intelligence, "Connected vehicle market: 2021 - 26: Industry share, size, growth - mordor intelligence," Aug 2020. [Online]. Available: https://www.mordorintelligence.com/industry-reports/connected-vehicle-market

[3] W. Yan, "A two-year survey on security challenges in automotive threat landscape," 10 2015, pp. 185–189.

[4] F. Kohnhäuser, D. Püllen, and S. Katzenbeisser, "Ensuring the safe and secure operation of electronic control units in road vehicles," in *2019 IEEE Security and Privacy Workshops (SPW)*, 2019, pp. 126–131.

[5] M. S. U. Alam, S. Iqbal, M. Zulkernine, and C. Liem, "Securing vehicle ecu communications and stored data," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6.

[6] F. Sagstetter, M. Lukasiewycz, S. Steinhorst, M. Wolf, A. Bouard, W. Harris, S. Jha, T. Peyrin, A. Poschmann, and S. Chakraborty, "Security challenges in automotive hardware/software architecture design," 01 2013, pp. 458–463.

[7] A. Singh and L. Singh, "Connected car market size, growth, trends, outlook 2021-2027." [Online]. Available: https://www.alliedmarketresearch.com/connected-car-market

[8] M. A., C. C., and D. McGehee, "Vehicle owners' experiences with and reactions to advanced driver assistance systems," AAA Foundation for Traffic Safety, Publication, 09 2018.

[9] X. Yin, J. Tan, and L. Li, "Development of a real-time monitoring system for ecu based on can bus," in *2010 2nd International Conference on Industrial and Information Systems*, vol. 1, 2010, pp. 273–276.

[10] U. Nations, 2021.

[11] V. Q. W. G. . . A. SIG, "Automotive spice process assessment / reference model," Nov 2017.

[12] K. Daimi, M. Saed, S. Bone, and J. Robb, "Securing vehicle's electronic control units," in *ICNS 2016*, 2016.

[13] "Road vehicles — Cybersecurity engineering," International Organization for Standardization, Geneva, CH, Standard, 2021.

[14] T. Cuatto, C. Passerone, C. Sansoe, F. Gregoretti, A. Jurecska, and A. Sangiovanni-Vincentelli, "A case study in embedded systems design: An engine control unit," *Design Autom. for Emb. Sys.*, vol. 6, pp. 71–88, 09 2000.

[15] M. Wolf and T. Gendrullis, "Design, implementation, and evaluation of a vehicular hardware security module," in *Information Security and Cryptology - ICISC 2011*, H. Kim, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 302–318.

[16] Z. Levi, "understanding-automotive-cyber-security-in-vehicle-network-ivn," Aug 2021. [Online]. Available: https://blog.nng.com/understanding-automotive-cyber-security-in-vehicle-network-ivn/

[17] C. Wilwert, N. Navet, Y.-Q. Song, and F. Simonot-Lion, "Design of automotive x-by-wire systems," 01 2005.

[18] R. B. GmbH, "Can specification 2.0," Sep 1991.

[19] O. Avatefipour and H. Malik, "State-of-the-art survey on in-vehicle network communication (can-bus) security and vulnerabilities," *ArXiv*, vol. abs/1802.01725, 2018.

[20] H. Malik, O. Avatefipour, A. Hafeez, and P. Raj, "Comparative study of can-bus and flexray protocols for in-vehicle communication," 04 2017.

[21] B. Vinodh Kumar and J. Ramesh, "Automotive in vehicle network protocols," in *2014 International Conference on Computer Communication and Informatics*, 2014, pp. 1–5.

[22] A. Grzemba, *Most: The Automotive Multimedia Network*. Franzis, 20011.

[23] T. Steinbach, *Ethernet-basierte Fahrzeugnetzwerkarchitekturen für zukünftige Echtzeitsysteme im Automobil*, 10 2018.

[24] S. International, "Ethernet backbones will support many branches," *Autonomous Vehicle Engineering*, p. 22–24, Sep 2021.

[25] A. Saxena, "Understanding autosar and its applications in the automotive industry," Apr 2020. [Online]. Available: https://www.einfochips.com/blog/autosar-in-automotive-industry/

[26] M. Staron and D. Durisic, *AUTOSAR standard*, 06 2017, pp. 81–116.

[27] A. d. cooperation, "Classic platform," 2021. [Online]. Available: https://www.autosar.org/standards/classic-platform/

[28] AUTOSAR, "Utilization of crypto services," Nov 2021.

[29] J. N. Bajpai, "Emerging vehicle technologies & the search for urban mobility solutions," *Urban, Planning and Transport Research*, vol. 4, no. 1, pp. 83–100, 2016. [Online]. Available: https://doi.org/10.1080/21650020.2016.1185964

[30] K. Bengler, K. Dietmayer, B. Farber, M. Maurer, C. Stiller, and H. Winner, "Three decades of driver assistance systems: Review and future perspectives," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 4, pp. 6–22, 2014.

[31] "How to read the stars: Euro ncap." [Online]. Available: https://www.euroncap.com/en/about-euro-ncap/how-to-read-the-stars/

[32] G. Sikander and S. Anwar, "Driver fatigue detection systems: A review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2339–2352, 2019.

[33] A. Moujahid, M. ElAraki Tantaoui, M. D. Hina, A. Soukane, A. Ortalda, A. ElKhadimi, and A. Ramdane-Cherif, "Machine learning techniques in adas: A review," in *2018 International Conference on Advances in Computing and Communication Engineering (ICACCE)*, 2018, pp. 235–242.

[34] M. Zhai, "Advanced driver assistant system threats, requirements, security solutions," 2015.

[35] V. K. Kukkala, J. Tunnell, S. Pasricha, and T. Bradley, "Advanced driver-assistance systems: A path toward autonomous vehicles," *IEEE Consumer Electronics Magazine*, vol. 7, no. 5, pp. 18–25, 2018.

[36] Mercedes-Benz, "Benz s-class: Intelligent drive next level." Jan 2021. [Online]. Available: https://www.mercedes-benz.com/en/innovation/autonomous/the-new-s-class-intelligent-drive-next-level/

[37] A. Ziębiński, R. Cupek, D. Grzechca, and L. Chruszczyk, "Review of advanced driver assistance systems (adas)," vol. 1906, 11 2017, p. 120002.

[38] D. Amara, N. Chebrolu, V. Ravi, and S. Kp, "A brief survey on autonomous vehicle possible attacks, exploits and vulnerabilities," 10 2018.

[39] W. Cunningham, "Us requiring back-up cameras in cars by 2018," Mar 2014. [Online]. Available: https://www.cnet.com/roadshow/news/u-s-requiring-back-up-cameras-in-cars-by-2018/

[40] A. S. Ahire, "Night vision system in bmw," *International Review of Applied Engineering Research*, vol. 4, no. 1, pp. 1–10, 2014.

[41] F. Castanedo, "A review of data fusion techniques," *TheScientificWorldJournal*, vol. 2013, p. 704504, 01 2013.

[42] M. S. Sheikh, J. Liang, and M. A. Khan, "A comprehensive survey on vanet security services in traffic management system," *Wirel. Commun. Mob. Comput.*, vol. 2019, jan 2019. [Online]. Available: https://doi.org/10.1155/2019/2423915

[43] T. Neudecker, N. An, O. Tonguz, T. Gaugel, and J. Mittag, "Feasibility of virtual traffic lights in non-line-of-sight environments," 06 2012.

[44] "Ieee 1609.12-2019 - ieee standard for wireless access in vehicular environments (wave) - identifiers," Oct 2019. [Online]. Available: https://standards.ieee.org/standard/1609_12-2019.html

[45] M. R. Ghori, K. Z. Zamli, N. Quosthoni, M. Hisyam, and M. Montaser, "Vehicular ad-hoc network (vanet): Review," in *2018 IEEE International Conference on Innovative Research and Development (ICIRD)*, 2018, pp. 1–6.

[46] P. Sewalkar and J. Seitz, "Vehicle-to-pedestrian communication for vulnerable road users: Survey, design considerations, and challenges," *Sensors*, vol. 19, no. 2, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/2/358

[47] A. Al-Ali, F. Aloul, N. Aji, A. Al-Zarouni, and N. Fakhro, "Mobile rfid tracking system," 04 2008, pp. 1 – 4.

[48] M. A. Rahim, M. A. Rahman, M. M. Rahman, A. T. Asyhari, M. Z. A. Bhuiyan, and D. Ramasamy, "Evolution of iot-enabled connectivity and applications in automotive industry: A review," *Veh. Commun.*, vol. 27, p. 100285, 2021.

[49] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, "Connected vehicles: Solutions and challenges," *IEEE Internet of Things Journal*, vol. 1, no. 4, pp. 289–299, 2014.

[50] S. Chen, J. Hu, Y. Shi, and L. Zhao, "Lte-v: A td-lte-based v2x solution for future vehicular network," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 997–1005, 2016.

[51] D. Kombate and Wanglina, "The internet of vehicles based on 5g communications," in *2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2016, pp. 445–448.

[52] F. Yang, J. Li, T. Lei, and S. Wang, "Architecture and key technologies for internet of vehicles: a survey," *Journal of Communications and Information Networks*, vol. 2, pp. 1–17, 2017.

[53] Y. Ai, M. Peng, and K. Zhang, "Edge cloud computing technologies for internet of things: A primer," *Digital Communications and Networks*, vol. 4, 07 2017.

[54] G. Kim and I. Jung, "Integrity assurance of ota software update in smart vehicles," *International Journal on Smart Sensing and Intelligent Systems*, vol. 12, pp. 1–8, 01 2019.

[55] J. Howden, L. Maglaras, and M. A. Ferrag, "The security aspects of automotive over-the-air updates," *International Journal of Cyber Warfare and Terrorism (IJCWT)*, vol. 10, pp. 64–81, 02 2020.

[56] K. Korosec, "How automakers will save 35 billion by 2022," Sep 2015. [Online]. Available: https://fortune.com/2015/09/04/ihs-auto-software/

[57] T. Chowdhury, E. Lesiuta, K. Rikley, C.-W. Lin, E. Kang, B. Kim, S. Shiraishi, M. Lawford, and A. Wassyng, "Safe and secure automotive over-the-air updates," 09 2018, pp. 172–187.

[58] J. Golson, "Many lexus navigation systems bricked by over-the-air softwarenbsp;update," Jun 2016. [Online]. Available: https://www.theverge.com/2016/6/7/11879860/lexus-navigation-broken-software-update-bug

[59] J. Barkai, "Vehicle diagnostics - are you ready for the challenge?" 10 2001.

[60] ISO, "Iso 14229-1:2020 uds," Feb 2020. [Online]. Available: https://www.iso.org/standard/72439.html

[61] P. Kharche, M. Murali, and G. Khot, "Uds implementation for ecu i/o testing," in *2018 3rd IEEE International Conference on Intelligent Transportation Engineering (ICITE)*, 2018, pp. 137–140.

[62] G. Nica, "Video: Here's how the bmw remote diagnosis feature works," Nov 2020. [Online]. Available: https://www.bmwblog.com/2020/11/30/bmw-remote-diagnosis-feature/

[63] U. Shafi, A. Safi, A. Shahid, S. Ziauddin, and M. Saleem, "Vehicle remote health monitoring and prognostic maintenance system," *Journal of Advanced Transportation*, vol. 2018, pp. 1–10, 01 2018.

[64] E. Nasr, E. F. Kfoury, and D. J. Khoury, "An iot approach to vehicle accident detection, reporting, and navigation," *2016 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)*, pp. 231–236, 2016.

[65] P. Shreyas, R. Roopalakshmi, K. Kari, R. Pavan, P. Kirthy, and P. Spoorthi, *IoT-Based Framework for Automobile Theft Detection and Driver Identification: ICCNCT 2018*, 01 2019, pp. 615–622.

[66] K. Böhm, D. Paula, and H.-G. Schweiger, "Event data recorder (edr) for accidents with automated vehicles," 11 2019.

[67] G. Inc, "Definition of autonomous vehicles - gartner information technology glossary." [Online]. Available: https://www.gartner.com/en/information-technology/glossary/autonomous-vehicles

[68] M. Hartwig, "Self-driving and cooperative car," Jan 2020.

[69] R. Furchgott, "Public streets are the lab for self-driving experiments," Dec 2021. [Online]. Available: https://www.nytimes.com/2021/12/23/business/tesla-self-driving-regulations.html

[70] SAE, "Sae standards news: J3016 automated-driving graphic update," May 2020. [Online]. Available: https://www.sae.org/news/2019/01/sae-updates-j3016-automated-driving-graphic

[71] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, and S. Savage, "Experimental security analysis of a modern automobile," in *2010 IEEE Symposium on Security and Privacy*, 2010, pp. 447–462.

[72] AUTOCRYPT, "Top 6 cybersecurity challenges unique to the automotive industry," Oct 2021. [Online]. Available: https://autocrypt.io/top-6-cybersecurity-challenges-automotive-industry/

[73] QPERIOR, "Unece," Mar 2021. [Online]. Available: https://www.q-perior.com/fokusthema/unece-wp-29-ein-game-changer-fuer-die-automobilindustrie/

[74] M. Oguchi and M. Fuse, "Regional and longitudinal estimation of product lifespan distribution: a case study for automobiles and a simplified estimation method." *Environmental science and technology*, vol. 49 3, pp. 1738–43, 2015.

[75] B. Turban, C. Wolff, A. Tsakpinis, and M. Kucera, "A decision model for managing and communicating resource restrictions in embedded systems design," in *2008 International Workshop on Intelligent Solutions in Embedded Systems*, 2008, pp. 1–12.

[76] L. Schnieder and R. Hosse, *Leitfaden Automotive Cybersecurity Engineering - Absicherung vernetzter Fahrzeuge auf dem Weg zum autonomen Fahren*, 11 2018.

[77] J. P. Monteuuis, A. Boudguiga, J. Zhang, H. Labiod, A. Servel, and P. Urien, "Sara: Security automotive risk analysis method," *Proceedings of the 4th ACM Workshop on Cyber-Physical System Security*, 2018.

[78] G. Macher, E. Armengaud, E. Brenner, and C. Kreiner, "A review of threat analysis and risk assessment methods in the automotive context," 09 2016, pp. 130–141.

[79] T. Ucedavélez and M. Morana, *Risk Centric Threat Modeling: Process for Attack Simulation and Threat Analysis*, 05 2015.

[80] Microsoft, "The stride threat model," 2005. [Online]. Available: https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20)

[81] A. Lautenbach, "Healing vulnerabilities to enhance software security and safety," Mar 2016.

[82] G. Macher, H. Sporer, R. Berlach, E. Armengaud, and C. Kreiner, "Sahara: A security-aware hazard and risk analysis method," 03 2015.

[83] B. SChneier, "Attack trees," *Dr. Dobb's Journal*, Dec 1999.

[84] G. Macher, C. Schmittner, O. Veledar, and E. Brenner, *ISO/SAE DIS 21434 Automotive Cybersecurity Standard - In a Nutshell*, 09 2020, pp. 123–135.

[85] "Cybersecurity Guidebook for Cyber-Physical Vehicle Systems," Society of Automotive Engineers, USA, SURFACE VEHICLE RECOMMENDED PRACTICE, 2016.

[86] BlackBerry, "Wp.29 cybersecurity vehicle regulation compliance." [Online]. Available: https://blackberry.qnx.com/en/ultimate-guides/wp-29-vehicle-cybersecurity#what-is-wp-29

[87] UNECE, "Uniform provisions concerning the approval of vehicles with regards to cyber security and cyber security management system," United Nations, Standard E/ECE/TRANS/505/Rev.3/Add.154, 01 2021.

[88] ——, "Uniform provisions concerning the approval of vehicles with regards to software update and software updates management system," United Nations, Standard E/ECE/TRANS/505/Rev.3/Add.155, 01 2021.

[89] Auto-ISAC, "Best practices – auto-isac," 2016. [Online]. Available: https://automotiveisac.com/best-practices/

[90] JasPar, "About jaspar." [Online]. Available: https://www.jaspar.jp/en/about_us

[91] A. Oldehoeft, "Foundations of a security policy for use of the national research and educational network," 1992-02-01 1992.

[92] "Road vehicles — Functional safety," International Organization for Standardization, Geneva, CH, Standard, 2018.

[93] S. Brunner, J. Roder, M. Kucera, and T. Waas, "Automotive e/e-architecture enhancements by usage of ethernet tsn," in *2017 13th Workshop on Intelligent Solutions in Embedded Systems (WISES)*, 2017, pp. 9–13.

[94] M. Ihle and B. Glas, "Impact of demonstrated remote attacks on security of connected vehicles," in *Fahrerassistenzsysteme 2016*, R. Isermann, Ed. Wiesbaden: Springer Fachmedien Wiesbaden, 2018, pp. 101–117.

[95] B. Mashadi and D. Crolla, "Vehicle powertrain systems," *Vehicle Powertrain Systems*, 11 2011.

[96] G. S. Buthker, "Automated vehicle electronic control unit (ecu) sensor location using feature-vector based comparisons," 2019.

[97] Bosch, "Park assist." [Online]. Available: https://www.bosch-mobility-solutions.com/en/solutions/parking/parking-assist/

[98] R. Ma, "Automotive Adaptive Front-lighting System Reference Design," Texas Instruments, Texas, USA, Reference design, 2013.

[99] "ITS Security - Threat, Vulnerability and Risk Analysis (TVRA)," European Telecommunications Standards Institute, France, Technical Report, 2017.

[100] "Information technology — Security techniques — Privacy framework," International Organization for Standardization, Geneva, CH, Standard, 2011.

[101] ISO, "Information technology — security techniques — methodology for it security evaluation," International Organization for Standardization, Standard ISO/IEC 18045:2008, 08 2008.

[102] A. Ruddle, B. Weyl, S. Idrees, Y. Roudier, M. Friedewald, T. Leimbach, A. Fuchs, S. Gürgens, O. Henninger, R. Rieke, M. Ritscher, H. Broberg, L. Apvrille, R. Pacalet, and G. Pedroza, "Security requirements for automotive on-board networks based on dark-side scenarios. deliverable d2.3: Evita. e-safety vehicle intrusion protected applications," *Fraunhofer ISI*, 01 2009.

[103] N. Bißmeyer, M. Feiri, A. Giannetsos, F. Kargl, M. Moser, M. J. Khodaei, and H. Jin, "Preparing secure vehicle-to-x communication systems," 2011.

[104] NHTSA, "Cybersecurity best practices for the safety of modern vehicles," National Highway Traffic Safety Administration, Notice 86 FR 2481, 12 2021.

[105] P. E. Black, M. L. Badger, B. Guttman, and E. N. Fong, "Dramatically reducing software vulnerabilities: Report to the white house office of science and technology policy," 2016.

[106] O. Top 10, "A06:2021 – vulnerable and outdated components," 2021. [Online]. Available: https://owasp.org/Top10/A06_2021-Vulnerable_and_Outdated_Components/

[107] E. Barker, "Recommendation for key management: part 1 - general," National Institute of Standards and Technology, Standard https://doi.org/10.6028/NIST.SP.800-57pt1r5, 05 2020.

[108] J.-D. Gadina, "x86-64 machine code obfuscation," Jul 2017. [Online]. Available: https://xs-labs.com/en/blog/2017/11/07/x86-64-machine-code-obfuscation/

[109] N. Bißmeyer, "Security in ecu production," 2016.

[110] G. Sgayou, "Jailbreaking subaru starlink," Aug 2020. [Online]. Available: https://github.com/sgayou/subaru-starlink-research/blob/master/doc/README.md

[111] D. E. Metzker, "Technical article: No cybersecurity without key security," Oct 2019. [Online]. Available: https://www.vector.com/int/en/news/news/technical-article-no-cybersecurity-without-key-security

[112] escrypt, "Implementing chains of trust with cycurhsm," 2019.

[113] SAE, "Hardware protected security for ground vehicles," February 2020. [Online]. Available: https://www.sae.org/standards/content/j3101_202002/

[114] . Turan, Barker, "Nist special publication 800-90b recommendation for the entropy sources used for random bit generation," 01 2018.

[115] AUTOSAR, "Requirements on intrusion detection system," AUTomotive Open System ARchitecture, Standard 976, 11 2020.

[116] D. E. Metzker, "Reliably detecting and defending against attacks," Mar 2020.

[117] R. Buskey and B. Frosik, "Protected jtag," in *2006 International Conference on Parallel Processing Workshops (ICPPW'06)*, 2006, pp. 8 pp.–414.

[118] "Ieee standard for test access port and boundary-scan architecture," *IEEE Std 1149.1-2013 (Revision of IEEE Std 1149.1-2001)*, pp. 1–444, 2013.

[119] AUTOSAR, "Specification of secure onboard communication," AUTomotive Open System ARchitecture, Standard 654, 12 2017.

[120] Quesnelle, "Is car hacking over? autosar secure onboard communication," April 2018. [Online]. Available: https://www.rsaconference.com/library/Presentation/USA/2018/is-car-hacking-over-autosar-secure-onboard-communication-2

[121] S. Sanwald, L. Kaneti, M. Stöttinger, and M. Böhner, "Secure boot revisited: Challenges for secure implementations in the automotive domain," *SAE International Journal of Transportation Cybersecurity and Privacy*, vol. 2, 08 2020.

[122] K. Schramm and M. Wolf, "Secure feature activation," *SAE International Journal of Passenger Cars - Electronic and Electrical Systems*, vol. 2, pp. 62–67, 2009.

[123] Vector, "Vector security manager for secure diagnostics," Apr 2019. [Online]. Available: https://cdn.vector.com/cms/content/events/2019/vSES19/vSES19_07_Lohmann_Vector.pdf

[124] ISO, "Road vehicles — diagnostic communication over internet protocol (doip)," International Organization for Standardization, Standard ISO 13400-2:2019, 12 2019.

[125] A. Weimerskirch, "Secure software flashing," vol. 1906, 01 2009.

[126] A. Greenberg, "Tesla responds to chinese hack with a major security upgrade," Sep 2016. [Online]. Available: https://www.wired.com/2016/09/tesla-responds-chinese-hack-major-security-upgrade/

[127] T. Karthik, Kuppusamy, and D. McCoy, "Uptane : Securing software updates for automobiles," 2016.

[128] A. Palanca, E. Evenchick, F. Maggi, and S. Zanero, "A stealth, selective, link-layer denial-of-service attack against automotive networks," in *Detection of Intrusions and Malware, and Vulnerability Assessment*, M. Polychronakis and M. Meier, Eds. Cham: Springer International Publishing, 2017, pp. 185–206.

[129] H.-K. Kong, T.-S. Kim, and M.-K. Hong, "A security risk assessment framework for smart car," in *2016 10th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*, 2016, pp. 102–108.