SAFURE

D1.1
Use Cases Specifications

Project number:	644080
Project acronym:	**SAFURE**
Project title:	SAFety and security by design for interconnected mixed-critical cyber-physical systems
Project Start Date:	1st February, 2015
Duration:	36 months
Programme:	H2020-ICT-2014-1
Deliverable Type:	Report
Reference Number:	ICT-644080-D1.1
Work Package:	WP 1
Due Date:	JUL 2015 - M6
Actual Submission Date:	3rd August, 2015
Responsible Organisation:	ESCR
Editor:	Lena Steden
Dissemination Level:	PU
Revision:	00.02
Abstract:	This deliverable defines three scenarios in the areas of telecommunication, automotive multi-core, and automotive network. It is the aim of the SAFURE project to develop a safe and secure design for interconnected systems in these industry areas. The developed design will be applied to these three scenarios which have various use cases.
Keywords:	use cases, scenarios, automotive, telecommunication, multi-core, network architecture

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 644080.

This work is supported (also) by the Swiss State Secretariat for Education, Research and Innovation (SERI) under contract number 15.0025. The opinions expressed and arguments employed herein do not necessarily reflect the official views of the Swiss Government.

Editor

Lena Steden(ESCR)

Contributors (ordered according to beneficiary numbers)

Christina Petschnigg (TEC)
Stefania Botta, Luigi Santamato (MAG)
Carolina Reyes (TTT)
Mikalai Krasikau (SYSG)
Jonas Diemer (SYM)
Sylvain Girbal (TRT)
Daniel Thiele (TUBS)
Jaume Abella (BSC)
Marco Di Natale (SSSA)
Philipp Miedl (ETHZ)
Dominique Ragot (TCS)

Disclaimer
The information in this document is provided "as is", and no guarantee or warranty is given that the information is fit for any particular purpose. The users thereof use the information at their sole risk and liability.

Executive Summary

This deliverable provides the specifications of three scenarios which will be realized in the SAFURE project. The telecom scenario, the automotive multi-core scenario, and the automotive network scenario.
For each of the three scenarios, the motivation and objective is given. The details of the scenario are presented in several use cases. It is described which actors and conditions are expected to be relevant in the use case.
The results of Task T1.1, on which this deliverable is based, are essential to create a common understanding of the project for all stakeholders. The specifications of use cases and scenarios will be a guideline during the implementation of demonstrators and, finally, the project results are going to be evaluated against these domain-specific use cases, and the requirements presented in D1.2.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This deliverable provides the specifications of three scenarios consisting of several use cases which are the basis of further work in the SAFURE project. The first scenario focuses on an application in the telecommunication area while the other two scenarios cover the automotive domain. The scenarios split into use cases as follows:

1. Telecom Scenario (cf. Section 3)
   - Body Area Network Cybersecurity
   - Improving security of Bring Your Own Device (BYOD), Choose Your Own Device (CYOD) and other variants
2. Automotive Multi-Core Scenario (cf. Section 4)
   - Modeling Safe & Secure software components in a multi-core powertrain (PWT) Electronic Control Unit (ECU)
   - Timing Analysis of software component allocation on multi-core ECU
   - Data Protection on multi-core ECU
   - Data Integrity on multi-core ECU
   - Mixed-critical ECU
   - Normal Operation
   - Safety-Faulty Mode
   - Safety-Faulty Mode
3. Automotive Network Scenario (cf. Section 5)
   - Fault Tolerance
   - Hardware Failure Tolerance
   - Attack Prevention
   - Admission Control and Network Reconfiguration
   - Inter-domain Traffic

In the **telecommunication** scenario, two applications of a safety and security critical system on mobile devices are presented. The challenges of health-monitoring devices, in this case a Body Area Network (BAN), and BYOD scenarios are presented. It is necessary that user data is processed in a way that provides integrity protection and to some extend confidentiality to ensure privacy. Furthermore, there are reliability, safety, and timing constraints to the presented applications.
In the **automotive** scenarios, two mixed-critical systems are described. The multi-core platform combines applications of different Automotive Safety Integrity Levels (ASILs) on one ECU. In addition

to the challenges of a system with various safety levels, security requirements, especially with the objective to ensure data integrity and confidentiality, apply. The use cases of the multi-core scenario cover the development phase of the platform as well as the expected normal operation mode and two error modes.
In the **automotive network** scenario, an Ethernet network with traffic of various priorities and real-time requirements is examined. Apart from fault and failure tolerance, attack prevention mechanisms shall be implemented.

1.1 Objectives of D1.1

The main objective of deliverable D1.1 is to define a set of use cases for both target industry domains, telecommunication and automotive industry. The SAFURE project has three objectives which have been defined in the Description of Action (DoA) [10]. It aims at implementing an holistic approach to safety and security by construction of embedded systems (**OBJ1**), it shall empower designers and developers with methods and tools to consider safety and security as well as communication and runtime system requirements (**OBJ2**) and it is an opportunity to extend current standards that will set the ground for the development of SAFURE-compliant safe and secure mixed-critical embedded products (**OBJ3**).
Deliverable D1.1 emerges from Task T1.1 in which industrial use cases have been specified. Three domain-specific scenarios with several use cases are described on a level that allows the identification of safety and security requirements. Demonstration, application, and evaluation of project results are done based on these domain-specific use cases.

1.2 Use of the D1.1 Outcomes

The scenarios and use cases specified in deliverable D1.1 are an important basis for other deliverables and work packages of the SAFURE project. The requirements specified by the partners in deliverable D1.2 are based on the use cases at hand. The work packages WP2, WP3, WP4, and WP5 aim at implementing and analyzing platforms that realize the defined use cases and fulfil the stated requirements. Finally, the implementations are going to be evaluated against the use case definitions in work package WP6. The dependencies between the different work packages are illustrated in Figure 1.1.

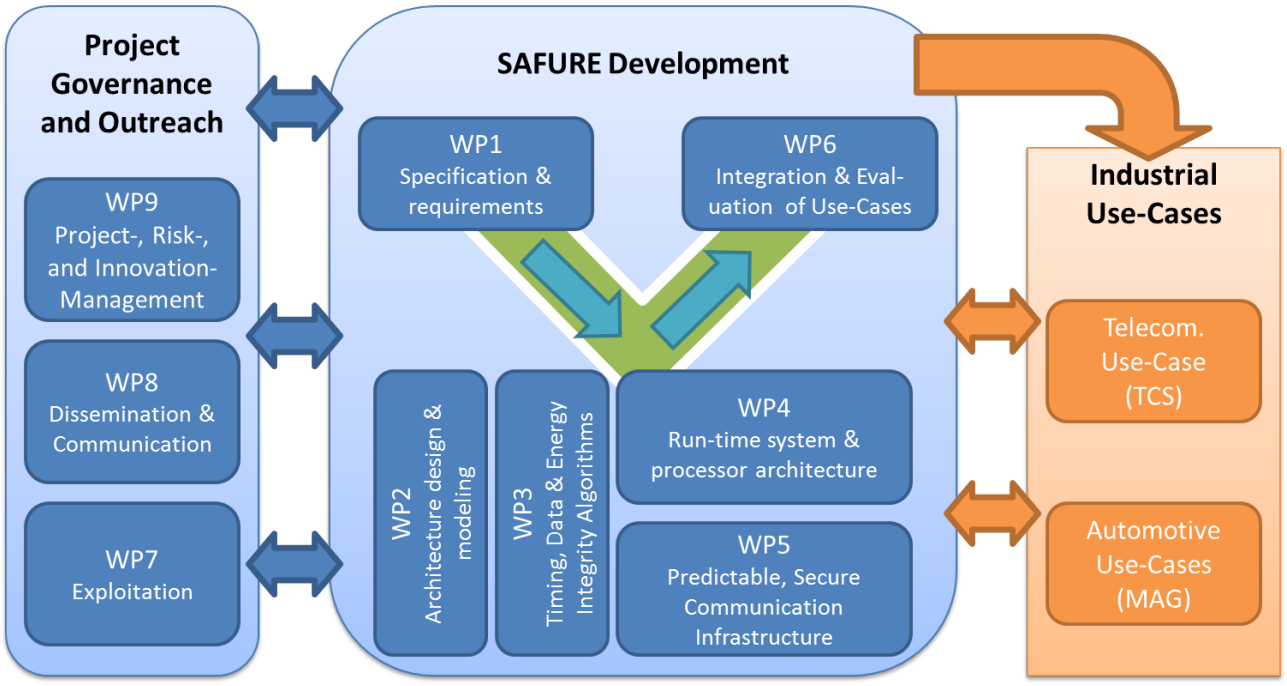

Figure 1.1: Workplan for SAFURE Project

Chapter 2

Methodology of Use Case Scenario Specification

In this chapter, the methodology of the use case scenario specification process is presented.

2.1 Use Case Scenario Development Process

The three scenarios presented in this deliverable, the telecommunication scenario, the automotive multi-core scenario, and the automotive network scenario, were proposed in the SAFURE Description of Action (DoA) [10]. For each scenario, a set of use cases has been derived by project partners who have profound knowledge and experience in the respective domains. The use cases model actions by users, characteristics of underlying platforms, and important stages in the development of the application.
In the deliverable, there is a chapter for each of the three scenarios. After giving a motivational scenario and the objectives of the scenario in the first two subsections, the use cases of each scenario are described in the third subsection. For most of the use cases, an overview of its key points is given in a table. The table contains the *actor*(s), i.e. which persons and/or system components are involved in the use case, the *goal*, i.e. what is achieved by this use case, the *trigger*(s), i.e. when is the use case started, as well as *preconditions* and *postconditions*, i.e. what is the status of the system before and after this use case. This table is helpful to get a first impression of the use case. Details are presented in the subsequent description paragraph.
After a first proposal, the scenario and use case descriptions were discussed with all partners involved in work package (WP)1 in a number of telephone conferences. Finally, an internal review has been conducted to ensure the quality and consistency of the deliverable. In this way, a set of meaningful use cases has been specified that will provide an important basis for the upcoming activities of the SAFURE project.

2.2 Justification and Selection of Scenarios and Use Cases

The industrial and academic partners agreed on working on a telecommunication and two automotive scenarios. Both domains raise **safety** and **security** requirements. Solutions need to cope with restricted hardware resources and - at the same time - have to be affordable and convenient for manufacturers and users. Therefore, the automotive and telecommunication domains were identified as challenging and promising areas of research.
In the **telecommunication** scenario, cf. Section 3, solutions will be developed to integrate a safety and security critical system on the limited hardware resources of a smart device. This industrial domain is undergoing rapid developments. Mobile infotainment systems and safety products, e.g. for health monitoring, are a growing market with many customers. Thus a reliable approach on how to

develop a safe and secure system on a smart device will be beneficial to many manufacturers and customers.
The **automotive multi-core** scenario, cf. Section 4, was chosen to show how a mixed-critical system can be implemented on a multi-core ECU. Additionally to the challenges of a system that covers mixed-critical safety level, data stored on the ECU shall be protected from disclosure and manipulation.
In the **automotive network** scenario, cf. Section 5, safety and security requirements will be implemented in an automotive network based on Ethernet. Apart from real-time communication and fault tolerance, the development of measures to prevent malicious traffic and other security threats are objectives of this scenario.

Chapter 3

Scenario 1: Telecom Scenario

Nowadays, smartphones are used for many applications including the growing development of social networks, cloud based applications, health apps and so on. In fact, the telecommunication industry evolved from a rich featured smartphone to a very powerful, multi-purpose tool, sitting most of the time right in our pocket.
Nevertheless, as the scope of smartphones uses keeps growing, new challenges arise. The pervasive use of smartphones and tablets for mixed-critical applications raises serious security challenges regarding the confidentiality, integrity and authenticity of data. As a unique endpoint is used with various independent infrastructure of different sensitivity, there is also a need for a partitioning scheme between these different domains. Data processed in a professional context should not be accessible from personal apps in order to ensure the confidentiality of professional information (mail, documents, etc.). In a similar way, integrity of medical applications can be critical for a patient under treatment and the device should provide a high assurance safety mechanism to handle health applications.

3.1 Motivational Scenario

All use cases identified share the same concerns about how to achieve mutual trust between various stakeholders involved in the creation, design, development, integration, administration, exploitation and, in the end, provisioning of the platform.
For each stakeholder, we must identify what and how the relevant data needs to be stored, integrated, shared or destroyed at the right moment, within the right context along with the associated process which must be implemented with the right assurance thought both the platform but also with the associated system. Therefore, one of the goals of this task is to identify the different stakeholders, agents, critical functions and sensitive objects which will handle relevant information and perform critical tasks.
While handling goods of various stakeholders on one hosted platform is nowadays a classical solution on smartphones or tablets, the scope is both narrowly and statically defined. It is limited to pre-integrated Intellectual Property (IP) protection or Digital Rights Management (DRM) enforcement over multimedia (movies, music, books, etc.) files. Furthermore, the few security protocols which can be used with the relevant assurance such as High-bandwidth Digital Content Protection (HDCP) are often pre-integrated into the platform but are clearly not open for other applications than those previously envisioned. Thus, only low assurance mechanisms are really available and do not entirely fit with the criticality of both goods (for example health data) and functionality (for example, control/command of medical devices). Furthermore, mobile classical multiple stakeholders mechanisms are platform oriented while the new topics envisioned in this scenario are mutually binding different systems (health, private sphere, industries) creating a de-facto low-integrated system of systems.

The multiple stakeholder (MS) paradigm is in fact, a double generalization of the CYOD paradigm. First, it is a generalization over the number of stakeholders since CYOD is a 2-tiers only implementation

of the MS paradigm while the MS paradigm can involves a larger number of stakeholders. Second, MS paradigm can also performs information processing over multi-tiers data through invocation of appropriate trusted subjects which are usually not present in CYOD where only seclusion and transfer mechanisms are available.

3.2 Objectives of Scenario

The following two use cases demonstrate how safety- and security-critical applications can be realized on smart devices. The telecommunication applications developed in this task will go through the modelling and analysis processes of WP2 and WP3 to provide a safe and secure result. A demonstrator is going to be implemented based on the requirements identified in deliverable D1.2 and the first ideas presented in deliverable D1.3. The final system is going to be evaluated against the safety and security requirements raised in WP1.

3.3 Description of the scenario and use cases

3.3.1 Use Case 1.1: Improving security of BYOD and CYOD

Introduction to BOYD and CYOD

When dual use (personal and professional) of smartphones began pervasive, security concerns acted as a major showstoppers. The first solution came from the USA with the Bring Your Own Device (BYOD) concept. The underlying idea of the BYOD is that collaborators install on their smartphone both the relevant software from their organization (usually a business operation) and a seclusion mechanism which separates the processing of business operation from other processes. Cultural and legal differences made BYOD not really relevant in Europe. Thus, Choose Your Own Device (CYOD) appeared as dual model (notably in Europe) where the smartphone is pre-installed with business software and data, and, a seclusion mechanism similar to the one used for BYOD, offers a zone dedicated for private data processing. The main difference between BYOD and CYOD is the ownership of the platform: In BYOD the user owns the platforms and installs business specific software and data, while in CYOD the organization owns the platform and delegates some private use to the user.
From both a legal and technical point of view, there is a huge difference between both solutions. BYOD requires to be available on almost any platform on the consumer market, thus interoperability requirements make the implementation of reliable security mechanisms extremely hard since these measures are usually rooted in the software stack. CYOD is a more useful paradigm in the context of SAFURE since the platform can be elected from a small set of available consumer platforms or refined from an available reference design. One can acknowledge that this is true if one wants to resolve dependability issues. Therefore, CYOD is the nearest model to SAFURE mixed-criticality requirements.
Many solutions exist to host both personal and professional environments in a single mobile device, here are some example:

- WorkPlay technology by InZero: `http://www.workplaytablet.com/`
- Good Suite by Good: `https://fr.good.com/applications/fr-emm-suites.html`
- Lacoon Mobile Security by Lacoon: `https://www.lacoon.com/`
- Capsule solution by Check Point: `http://www.checkpoint.com/products-solutions/mobile-security/check-point-capsule/index.html`
- TEOPAD by Thales: `https://www.thalesgroup.com/en/teopad-security-solution-smartphones-and-tablets`

- Knox by Samsung:
  `http://www.samsung.com/us/business/samsung-for-enterprise/samsung-knox.html`
- AimPoint

Note: Selected Samsung devices with the KNOX Workspace embedded received CC certification based on Mobile Device Fundamentals Protection Profile (MDFPP)[1]. The idea of this use case is to go beyond sandboxing and containerization, which are the most used technologies implemented in the previous solutions but which are unfortunately not flawless. The SAFURE approach is to secure an existing mobile device using a high assurance separation kernel to provide a security-enhanced CYOD solution.

Figure 3.1: Security vs. Versatility of Mobile Devices

Figure 3.1 illustrates the trade-off between security level (and, implied by this factor, cost) and versatility of mobile devices. While a commercial off-the-shelf (COTS) smartphone is versatile, it provides little to no security mechanisms. TEOPAD [12] is a mobile solution that allows customers to separate personal and professional content on one smartphone or tablet. This solution can be used on many platforms and provides minimum security measures. The TEOREM OIV [13] is located in the opposite corner of the diagram, offering a much higher level of security but less versatility. The SAFURE device envisioned for this telecommunication scenario is a trade-off between these two products: It is

[1] `https://www.samsungknox.com/en/faq/what-mdfpp`

versatile, has a number of interfaces for remote connections, and provides a sufficient level of security by a temper-resistant Secure Element (SE) and its separation kernel.

3.3.2 Use Case 1.2: Body Area Network Cybersecurity

Description

Body Area Network (BAN) is a wireless network at the human body scale which consists of wearable computing devices embedded inside the body, surface-mounted on the body in a fixed position or carried in different position in clothes pocket, by hand or in various bags [16]. The ongoing increase in the usage of BAN in various applications (e.g. in sport, healthcare, entertainment, man-to-machine) raises new challenges regarding safety, security and privacy. Most notably, balancing security, privacy, safety, and utility is a necessity in the health care domain. In modern medicine, medical devices embedded inside the human body, also called Implantable Medical Devices (IMDs), enable remote monitoring of a patient's health status [9]. They continuously and automatically monitor a number of health conditions. Modern medical devices and their software suffer from security problems induced by wireless capabilities in implantable medical devices. They are becoming part of the Internet of Things (IoT) and, in this way, these devices suffer from cybersecurity issues.

Addressing safety and security of a smartphone used as an insulin pump remote control

For many years, insulin therapy has been possible by using an insulin pump system rather that manually using syringes. This kind of system includes an insulin pump, a blood glucose monitor and a remote control. These components are interconnected and employ wireless communication to form a real-time monitoring and response loop. While insulin pump systems have significantly improved patients life, new safety and security risks have emerged. Many academic papers have been written on the topic of securing remote control devices for insulin pump or in general for IMDs [2, 6, 8]. The conclusion of these studies shows that security issues require the use of a dedicated hardware as a control and monitoring tool.
The idea of this use case is to extend the concept of BYOD for medical application. Instead of using a dedicated piece of hardware as a remote control or monitoring device, the patient can use his own smartphone as a control and command device for his insulin pump (or any other IMDs that require real-time monitoring). The phone shall still be able to run an operating system like Android so the user can install various apps available on the market.
This particular mobile device shall enforce the set of security requirements defined in the SAFURE framework and it is designed using the SAFURE methodology and tools.
The main risks are patient privacy loss due to data leakage from the device, inappropriate medical follow-up and device unavailability resulting in loss of IMD control. In other words, the challenge is to protect smartphone apps for calculating and controlling insulin dose to be delivered: If malicious code modifies the dose calculator, it puts users at risk of either catastrophic overdose or suboptimal glucose control. If the measured values of glucose are maliciously or unintentionally modified, the physician might unwittingly put the patient at risk by delivering an erroneous dose of insulin. Figure 3.2 depicts the separation between IMD control & monitoring tool and personal application.

Device overview and features

The following technologies are involved in use case 1.2:

- Device with the latest Android version while keeping the security at a sufficiently high level.
- Existing device using a high assurance separation kernel so that critical medical application can be executed.

Additional functionality:

Figure 3.2: Medical Use Case

- Such mobile devices could also offer remote access to the physician or other member of the medical team to access the history or to change settings.
- In addition, the system can be used to raise an alarm in case of pump malfunction or abnormal glucose evolution. The event shall be sent over the air to a dedicated server that is accessible by the physician. The security objective to address the network eavesdropping is cover in the MDFPP but in this situation requires also an availability of the system. So we introduce a new security objective for the system: The availability of both the mobile device in the infrastructure (the cloud).

Market Watch:

- In 2011, Debiotech released an android-powered device to control the insulin pump that uses a dedicated SIM card to ensure the security of the system. They have designed a controller app that is able to disable regular functions of the phone while making pump adjustments, when the pump mode is enabled, it is not possible to send or receive calls or emails [11]. As mobile devices and Operating Systems (OSs) evolve very fast, it would be convenient for the end user to merge the medical functionality with the everyday personal functionality into one device.
- At the end of 2014, the US Food and Drug Administration (FDA) release guidance relative to cybersecurity in medical devices [14].

In this use case, the assurance level for data integrity is clearly quite high. Privacy issues are also a concern in this kind of medical system as the personal mobile device of the patient, running applications from various sources, is storing blood glucose value. Installed or downloaded applications, which are potentially malicious, shall not have access to those values.

3.4 Summary

Table 3.1 gives an overview of the relation of the automotive network use cases to work packages and the SAFURE objectives.

Use Case	Type	Description	Workpackages	Objectives
1.1	security	Improving security of BYOD and CYOD	WP2, WP3, WP4, WP5, WP6	OBJ1, OBJ2, OBJ3
1.2	safety, security	Body Area Network Cybersecurity	WP2, WP3, WP4, WP5, WP6	OBJ1, OBJ2

Table 3.1: Use case relations to work packages and SAFURE objectives

Chapter 4

Scenario 2: Automotive Multi-Core Scenario

Automotive systems designers face constantly increasing demands for more performance and shorter time-to-market periods. Embedded processors need to perform an expanding set of tasks - often in real time. Meanwhile, applications demand high throughput and energy efficiency coupled with small form factors and low cost. Multicore Microcontroller Units (MCUs) provide a feasible new solution, leveraging modular design to deliver multi-fold performance increases at an economical price.
For decades, as the number of transistors on an Integrated Circuit (IC) increases, chip performance has kept pace. Ever more sophisticated architectures featuring techniques such as caching and pipelining allowed chip designers to use the increasing density of the silicon to continually boost processing speed. That is no longer the case. Chip designers have exhausted the possibilities of alternative architectures. The only way to increase productivity today is to leverage modularity by using multiple Central Processing Units (CPUs). That has led to the development of multi-core MCUs also for the automotive market.
Historically, many safety-related and security-critical systems have been developed and qualified using single-core processors. These platforms could easily meet increasing requirements regarding system performance requirements by higher processor clock speeds. However, the industry is now approaching the limit of relatively simple upgrade path, and there is a significant trend towards the adoption of multi-core processor architectures in critical systems to address higher performance demands.
Modern automotive systems are becoming increasingly complex and connected. Nowadays vehicles contain several dozens of ECUs with powerful processing capabilities, running different soft- and hardware architectures interconnected via various types of on-board network infrastructure. The traditional design of vehicles has - until recently focused on the standard functionality like engine management, steering, and breaking. The next step has been to include safety engineering (which has also in between reached a certain maturity including standard architectures like lockstep processing) aiming to increase the safety of modern vehicles to the latest norm requirements. Developments like anti-lock brake systems, redundant buses, and the general constant inclusion of safety-related elements in all automotive systems made vehicles much safer for their passengers and for other traffic participants. The publication of an automotive functional safety standard ISO26262 [5] in 2011 is a benchmark in the functional safety domain.
However, the entire functional safety domain is not considering the existence of any situation in which the generation of a fault is the result of an intentional, malicious attack. The assumed fault model is based on statistical physical defects and of unintended systematic faults induced during the development cycle. Independent from safety considerations, some security measures like tuning protection were introduced since the late nineties. Such protection measures shall prevent any modifications in the system which have not been authorized by the Original Equipment Manufacturer (OEM), e.g. engine performance upgrades or activation of features which are implemented in software but are not enabled on the particular ECU. Besides the traditional on-board communication networking based,

e.g. on different Controller Area Network (CAN) types or FlexRay, an increasing number of vehicles offers wireless interfaces ranging from infotainment Near-Field Communication (NFC) to telematics infrastructure services, such as automatic crash response and remote diagnostics via cellular networks. This accessibility of the vehicle from the outside creates a full range of new vulnerabilities and entry points for a potential attacker. Therefore, automotive security has recently become an important field of research and received a lot of focus from OEMs and the entire value chain of vehicle industrial production. The vulnerabilities shown in many studies of commercial available cars imply the immediate need of new techniques and measures for increasing the security level of the current automotive implementations.
Safety and security mechanisms will affect system performances, thus taking advantage from the migration to multi-core systems. Moreover some of the issues addressed by ISO-26262, like „freedom of interference“, can take a real advantage from the hardware separation provided by multi-core microcontrollers.

4.1 Motivational Scenario

This automotive multi-core use case has two sub-scenarios: Scenario A covers the development of mixed-critical multi-core software, while Scenario B deals with operating a mixed-critical multi-core system.
Scenario A deals with the phase of integrating two or more mixed-critical software components on the same hardware platform. The main actors are software designers and developers. In order to set up a mixed-critical system on a multi-core ECU, a carefully designed engineering process is necessary. The use cases will demonstrate how IP protection, secure CAN communication, and safety requirements can be taken into account during the development process to ensure a safe and secure implementation of a mixed-critical system on a multi-core platform.

Scenario B covers the different operational phases of a multi-core ECU. In the normal operation mode (Use Case 2.B1), the two components, Engine Control and Transmission Control, run in separate memory partitions and exchange information via CAN if necessary. The system switches into the safety-fault mode (Use Case 2.B2) if one of the components tries to access the memory area of another partition. The security-fault mode (Use Case 2.B3) is entered when one of the partition tries to send unauthenticated CAN messages or an external tool attempts to flash the code storage of the Engine Control. It is important that no safety threats arise from any of these modes.

4.2 Objectives of the Scenario

The following use cases will demonstrate how safety and security requirements can be fulfilled in a mixed-critical automotive set up on a multi-core ECU. This multi-core scenario follows a holistic approach: The use cases of Scenario A describe how safety and security can be ensured during the development process while Scenario B deals with safe and secure modes of operation of the ECU.
The automotive applications developed in this task will go through the modelling and analysis processes of WPs 2 and 3 to provide a safe and secure distributed deployment. A demonstrator is going to be implemented using the run-time enhancements which are a result of WP 4 on top of the open source ERIKA operating system for automotive applications. The case study is characterized by hard real-time requirements and the need of an extremely small OS overhead regarding space and time. The applications (engine control and/or a gearbox in conjunction with the calibration board) will be modelled and adapted to run on a multi-core.

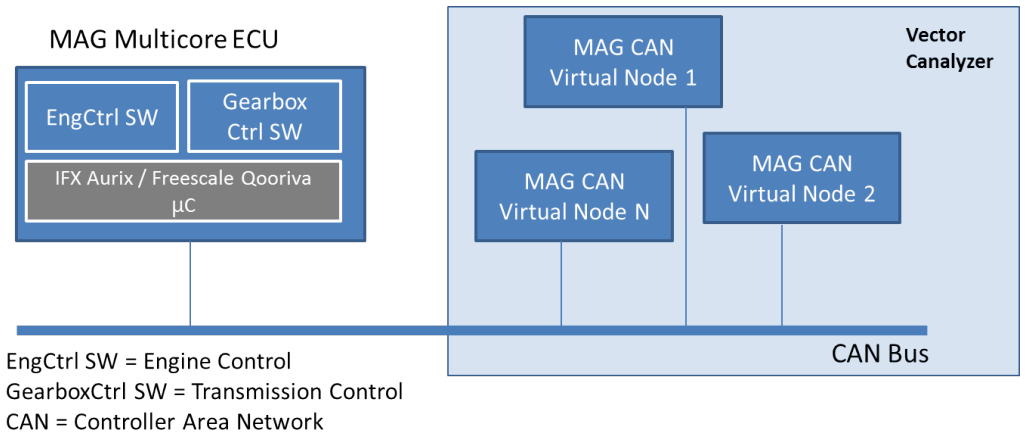

Figure 4.1: Environment of Automotive Multi-Core Use Cases

The environment of the demonstrator for all automotive multi-core scenarios is illustrated in Figure 4.1. This prototype of an automotive multi-core ECU will be based on either an AURIX microcontroller [4] by Infineon or Freescale's Qorivva MPC5777 [3] connected through the CAN bus to a PC with Vector CANalyzer [15] tool. The PC can simulate various virtual CAN nodes. The multi-core ECU will provide Engine Control software on Core 0 and Transmission Control software on Core 1 of the microcontroller. To achieve an holistic setup for both automotive scenarios, the demonstrator developed for the automotive multi-core scenario will be connected to the Ethernet bus of the network scenario by an CAN/Ethernet gateway as illustrated in Figure 4.2. The final system is going to be evaluated against the safety and security requirements raised in WP 1.

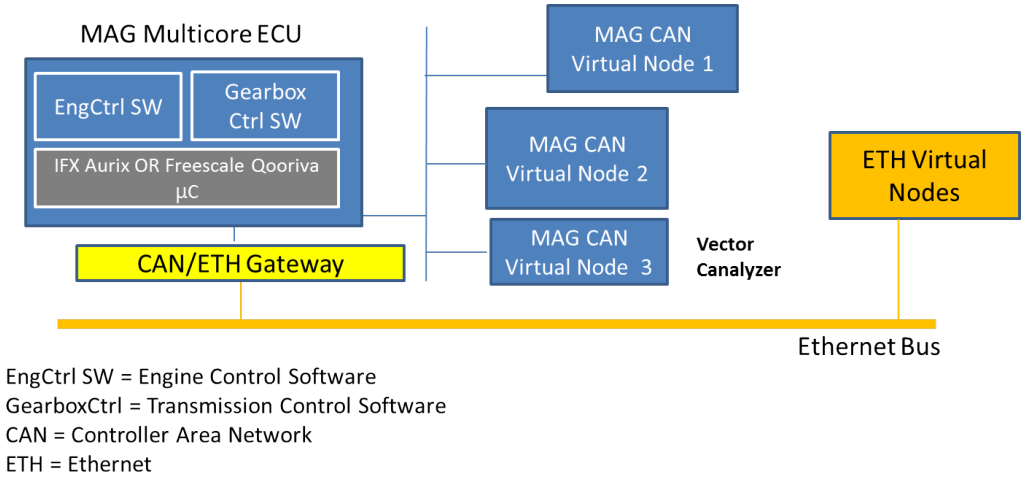

Figure 4.2: Integration of the automotive multi-core demonstrator in the network scenario

4.3 Description of the scenarios and use cases

4.3.1 Use Case 2.A1: Modelling Safe & Secure software components in a multi-core PWT ECU

Description

Actual software component models lack the link with hardware resources (core and peripherals), so it is not feasible to model a multi-core partitioning that is not affected by conflicts on hardware resources. Also AUTomotive Open System ARchitecture (AUTOSAR) [1] lacks on multi-core modelling guide-

lines. Also in actual automotive models it is not possible to model safety (like spatial and temporal isolation referred by ISO-26262 [5]) and security aspects of software components.

4.3.2 Use Case 2.A2: Timing Analysis of software component allocation on multi-core ECU

Description

Automotive use case lacks actually of timing analysis tools and algorithms to optimize allocation of software component on a multi-core environment, thus enforcing design part of actual ECU software.

4.3.3 Use Case 2.A3: Data Protection on multi-core ECU

Description

ECU software is entirely stored inside μC flash memory. Adversaries can read flash contents through μC debug interfaces and reverse-engineer the source code included the company's IP.
This use case will demonstrate how data protection mechanisms can be integrated on a multi-core ECU to provide IP protection.

4.3.4 Use Case 2.A4: Data Integrity on multi-core ECU

Description

Data stored on the ECU, i.e. software, configurations, and - if applicant - user data, can be modified via CAN.
This use case will demonstrate how messages can be exchanged over the CAN bus in a secure way, i.e. in a way that ensures data integrity and authenticity. It is closely linked to Use Case 2.B2 of the Automotive Network Scenario where methods to prevent network attacks are developed.

4.3.5 Use Case 2.A5: Mixed-critical ECU

Description

Currently, Engine Control and Transmission Control are realized on two separate ECUs in the PWT system.
This use case will demonstrate how the functionality of the two ECUs can be realized on the same device to reduce cost. The special safety requirements of the Engine and Transmission Control functionality must be covered by ensuring spatial and temporal isolation of the two parts. The use cases of Scenario B give a more detailed insight in what challenges arise from this integration.

4.3.6 Use Case 2.B1: Normal Operation

Actor	ECU Core 0, ECU Core 1
Goal	Provide Engine Control and Transmission Control functionality on one ECU.
Trigger	Vehicle is started
Precondition	Vehicle is turned off
Postcondition	System in normal operation mode; Engine Control and Transmission Control functionality is available

Table 4.1: Overview Use Case 2.B1: Normal Operation

Description

In Normal Operation Mode, both Engine Control and Transmission Control work normally and exchange required data as it happens on separate ECUs. Engine Control and Transmission Control must run in separate memory partitions and can share required information via CAN bus. Some authenticated traffic will be generated on CAN bus to exchange information with all other CAN nodes in the network. In this mode, the CPU load increase for Engine Control and Transmission Control application caused by safety and security mechanisms is less than 10%.
This use case will demonstrate how Engine Control and Transmission Control operate in a default setting on one multi-core ECU.

4.3.7 Use Case 2.B2: Safety-Faulty Mode

Actor	ECU Core 0, ECU Core 1
Goal	Manage read/write access to memory; Provide uninterrupted functionality in case of read/write attempts to the memory area of the other component
Trigger	Component requests read/write access to memory part that is assigned to another component.
Precondition	System in normal operation mode
Postcondition	Rejection of read/write access; Uninterrupted functionality of ECU

Table 4.2: Overview Use Case 2.B2: Safety-Faulty Mode

Description

The system enters the safety-faulty mode when one of the two components, Engine Control or Transmission Control, tries to access the others memory area. In this case, the request is denied and the operation fails. The requesting entity keeps track of the failure. It is important that the other entity is not aware of the unauthorized access attempt and continues its regular operations.
This use case will demonstrate how unauthorized access attempts on designated memory partitions can be detected and prevented without interrupting the regular operating of the components.

4.3.8 Use Case 2.B3: Security-Faulty Mode

Actor	ECU Core 0, ECU Core 1, ECU Flashing Tool
Goal	Detect unauthorized CAN messages; Provide uninterrupted functionality in case of unauthorized access via CAN bus
Trigger	Unauthorized CAN message
Precondition	System in normal operation mode
Postcondition	Rejection of unauthorized CAN message; Uninterrupted functionality of ECU

Table 4.3: Overview Use Case 2.B3: Security-Faulty Mode

Description

The system enters the security-faulty mode when unauthenticated CAN messages are sent on the bus. These messages are discarded. The Engine Control and Transmission Control component have access to cryptographic mechanisms to be able to exchange authenticated CAN messages. Additionally, this mode of operation is entered if a flashing tool attempts to update the memory partition of the Engine Control without authentication.
This use case will demonstrate how unauthenticated messages can be detected on the CAN bus and

how the two components can fulfil security requirements without violating constraints arising from safety requirements.

4.4 Summary

As identified in the Description of Action (DoA) [10], one of the SAFURE objectives is to explore safety and security by domain (shared resources, multi-core processors). The safety and security requirements of each application must be conserved when integrating them - both at the communication and computation level. The use-cases that have been identified for the automotive scenario will address safety and security aspects relevant in the automotive domain. The following table, cf. Table 4.4, gives an overview of the relation of the automotive multi-core use-cases to work packages and SAFURE objectives.

Use Case	**Type**	**Description**	**Work Packages**	**Objectives**
2.A1	safety, security	Modelling Safe & Secure software components in a multi-core PWT ECU	WP2, WP6	OBJ2, OBJ3
2.A2	timing	Timing Analysis of software component allocation on multi-core ECU	WP2, WP6	OBJ2
2.A3	safety	Data Protection on multi-core ECU	WP3, WP6	OBJ1
2.A4	security	Data Integrity on multi-core ECU	WP3, WP6	OBJ1
2.A5	safety	Mixed-critical ECU	WP4, WP6	OBJ1
2.B1	safety	ECU Normal Operation	WP6	OBJ1
2.B2	safety	Safety ECU Faulty Operation	WP6	OBJ1
2.B3	security	Security ECU Faulty Operation	WP6	OBJ1

Table 4.4: Use case relations to work packages and SAFURE objectives

Chapter 5

Scenario 3: Automotive Network Scenario

Under the pressure of increasing bandwidth demand from e.g. advanced driver assistance systems or infotainment, in-vehicle networks are currently reaching their limits and are expected to transition to an Ethernet-based backbone network. This backbone network interconnects domains with different timing and safety requirements, so that both critical, e.g. (closed loop) control, and non-critical (best effort/infotainment) traffic streams must safely share the same communication infrastructure. Hence, this communication infrastructure must provide freedom from interference or at least sufficient independence between mixed-critical traffic streams (as mandated by ISO 26262 [5]). In order to provide sufficient independence from interference and to avoid congestion in the network, a preventive admission control scheme can be utilized, which (dynamically) manages and configures the resources of the communication network, e.g. bandwidth, paths (routes), switch buffers, CAM tables, etc.
In order to save wiring costs, it is expected that multiple mutually exclusive driving scenarios, e.g. driving, parking) share network resources. In this case as well, an admission control scheme can be utilized to manage the mutual access of different scenarios to network resources.
Software Defined Networking (SDN) protocols like OpenFlow [7] have been developed for the purpose of configuring and monitoring the network infrastructure. A network admission control mechanism relies on these protocols to actually configure the network, e.g. request and release resources, configure ingress filters, etc. In safety-critical systems, additionally, the timeliness and robustness of these SDN protocols, e.g. in case of link faults or failures, must be guaranteed. The monitoring functionality of these protocols can also be used to detect failures or attacks, e.g. by ingress filtering/policing or by evaluating the switch status and switch performance counters.
Many safety-critical applications, e.g. steer-by-wire or highly automated/autonomous driving, require fail-operational fault and error handling such that errors do not escalate to network (or system) failures. A vehicular backbone network must, hence, provide a fail-operational infrastructure. Different fault handling/mitigation strategies can be employed to address various faults. Permanent faults (e.g. link, switch, end station faults) can be handled by redundancy or the (timely) setup of alternative paths, transient faults (e.g. packet loss, overload) can be handled by redundancy or (timely) retransmission. Setting up alternative paths or retransmission intrinsically requires a certain time to detect a faulty path and to setup the alternative. Furthermore, any attacks to the network (e.g. flooding with malicious traffic, unauthenticated traffic), which can thread freedom from interference, must be blocked or (at least) shaped for fault containment. Real-time systems have strict timing requirements on the worst-case delay of these fault and attack detection and recovery times. Some (less critical) traffic streams might tolerate occasional frame loss or deadline misses. This can be covered analytically by weakly hard constraints.
An environment comprising high end-to-end communication security is a requisite for advanced vehicular applications. Such an environment should guarantee that all transferred information is seen and received in clear only by the desired parties, that potential modifications in Hardware or Software

are impossible to conceal and that unauthorized parties are incapable of participating in vehicular communication. Unauthorized vehicle modifications may represent a threat for the driving safety of the respective car and of all surrounding road users. Modern communication security mechanisms providing manipulation prevention and authentication based on cryptographic methods for all data transmission are being researched. These methods have become a good alternative to tackle most of the vehicular security issues.

5.1 Motivational Scenario

The driver enters a vehicle and starts driving. During normal operation, all messages are delivered timely and within their safety margins (Use Case 3.1). At some point in time, a loose contact at one of the network links causes occasional faults, e.g. dropping of frames with invalid Cyclic Redundancy Check (CRC) sums (Use Case 3.2). The affected link transports both high-critical and less-critical traffic streams. The high-critical traffic streams are sent to their destination via redundant paths, such that the faults are detected, but they cannot cause errors, i.e. no frames are lost and all frames reach their destination in time. For less-critical traffic streams, relaxed safety and timing constraints apply: lost frames are detected and their retransmission is requested by an Automatic Repeat Request (ARQ) protocol or occasional frame loss is tolerated by weakly hard constraints. The fault is reported to the driver.
Eventually an Ethernet link, which transports high-critical, less critical, and best effort traffic streams, breaks (Use Case 3.3). The high-critical traffic streams are sent to their destination via redundant paths, such that the broken link does not escalate to a network failure. For less-critical traffic, after error detection, the network is timely reconfigured by a network controller via SDN mechanisms to use an alternative path avoiding the broken link. Network resources of best effort traffic streams are released to make room for the rerouted critical traffic streams. The link failure is reported to the driver.
The diver contacts a workshop the repair the vehicle. At the workshop, the personnel puts the vehicle into maintenance mode by requesting corresponding permissions and traffic stream reservations from the network's admission controller (Use Case 3.4). The workshop personnel repairs the vehicle and instructs the network controller to leave maintenance mode.
A user device is connected to the vehicle's network via an user accessible port (e.g. Ethernet socket or WiFi). This device receives a certain bandwidth constraint by the networks admission controller. In order to enforce freedom from interference for critical traffic streams, the network controller configures the network accordingly, including traffic monitoring at the switches. Malicious software on the device (e.g. a virus on a passenger's smart phone), runs a denial of service attack by trying to flood the network with garbage Ethernet frames, which are outside the device's negotiated bandwidth constraint. This attack is detected immediately (i.e. within a bounded time interval) by the switch connecting the user device and the violating traffic is blocked before it can affect other traffic streams (e.g. their timing integrity) in the network (Use Case 3.5). Another attack vector is a garage employee with full physical access to all transmission media and devices that can be affected in the automotive network. This person might want to break in into the network to: 1) attack the passenger's privacy (phone tapping, data theft), 2) establish well directed attacks on particular vehicle components in the case of a theft or 3) even plan a potential assault. Cryptographic methods are a tool to improve automotive bus communication security, for instance, in the presence of a man-in-the-middle attack (Use Case 3.5).

5.2 Objectives of Scenario

The following use cases will demonstrate different network mechanisms, which are required for time- and safety-critical automotive networks. These capabilities are mainly developed during WP5 (focus on Ethernet network scheduling). Some methods like typical case analysis, which are used to verify

the relaxed timing (modelled by weakly hard constraints) of soft real-time systems are investigated in WP3.

5.3 Description of the scenario and use cases

For all use cases, we assume that there is some sort of admission control and network control available in the network, in order to enforce freedom from interference and to re-configure the network, e.g. in case of component failures. This can, for example, be realized by SDN. Some kind of monitoring is required to detect misbehaving traffic streams and attacks. Traffic shaping or blocking mechanisms are required to mitigate/contain such traffic streams and attacks. We will evaluate the following use cases under different network loads, e.g. traffic from driving scenarios, defined by industrial partners.

5.3.1 Use Case 3.1: Inter-domain Traffic

Note

This is a shared use case between the automotive multi-core scenario and the automotive network scenario.

Actor	ECU
Goal	Timely and reliable ECU-to-ECU communication over Ethernet backbone network
Trigger	An ECU sends a message to another ECU via Ethernet
Precondition	Network in valid state
Postcondition	Message delivered within its timing bounds and safety requirements

Table 5.1: Overview Use Case 3.1: Inter-domain Traffic

Description

In the near future, Ethernet will be used as a in-vehicle backbone network. ECUs in various domains (e.g. power train), which use (legacy) buses like CAN or FlexRay, are connected to this backbone via special gateways. These gateways might be part of an ECU of a switch. In timing- and safety-critical systems, the message exchange between ECUs in different domains requires formal analysis to proof that messages are delivered timely and reliably. This requires a heterogeneous analysis of the entire system considering the communicating ECUs (including their software stacks), the gateways, and the Ethernet.
This use case will connect the automotive multi-core scenario and the automotive network scenario to enable and demonstrate end-to-end formal analysis coverage, e.g. the worst-case end-to-end latency from an ECU on a CAN bus in the power train domain via gateways and Ethernet to a second ECU. The main focus of this use case, however, is the interconnect (i.e. CAN buses, gateways, and Ethernet). If feasible, the analysis will be extended to cover the ECU part (software) as well.

5.3.2 Use Case 3.2: Fault Tolerance

Actor	Faulty component, e.g. link or switch
Goal	Provide fault tolerant service
Trigger	Frame drop
Precondition	Network in normal operation
Postcondition	No degradation of service offered by the network

Table 5.2: Overview Use Case 3.2: Fault Tolerance

Description

In order to provide fault tolerance against transient faults to certain traffic streams, ARQs and Typical Case Analysis (TCA) should be supported. An ARQ scheme initiates an automatic retransmission of missing (e.g. dropped or corrupted) frames. It, hence, ensures that all frames from a sender reach their designated receiver. TCA, on the other hand, can be used for soft real-time traffic streams, and uses weakly hard constraints to bound, for example, the number of allowed frame drops or deadline misses. For hard real-time traffic streams, which do not tolerate the additional delay of a retransmission and also do not tolerate missing frames or deadline misses, redundant paths, where a frame is sent via multiple paths to its destination, might be a suitable solution.
This use case will demonstrate these fault tolerance mechanisms along with formal guarantees regarding their timing properties within the Ethernet concept developed during the project.

5.3.3 Use Case 3.3: Hardware Failure Tolerance

Actor	Faulty component, e.g. link or switch
Goal	Provide uninterrupted service to critical traffic streams
Trigger	Component failure
Precondition	Network in normal operation
Postcondition	Continued network service for critical traffic streams

Table 5.3: Overview Use Case 3.3: Hardware Failure Tolerance

Description

Selected critical traffic streams require continued network service even in the presence of hardware (e.g. permanent) failures, e.g. broken links or switches. As in Use Case 3.2, this can be achieved by using redundant paths. An alternative is to reconfigure the network to re-route critical traffic around the broken component. This path setup must complete in bounded time. Note that we will investigate in WP5 if the alternative path setup is a feasible approach.
This use case will demonstrate that the mechanisms developed in the project can provide continued network service for time and safety critical traffic streams.

5.3.4 Use Case 3.4: Admission Control and Network Reconfiguration

Actor	Workshop personnel
Goal	Configure network for new traffic requirements
Trigger	A configuration request packet is sent to the admission controller
Precondition	Network in valid state
Postcondition	Network reconfigured and in valid state

Table 5.4: Overview Use Case 3.4: Admission Control and Network Reconfiguration

Description

This use case covers different scenarios:
Workshop: The workshop personnel requests maintenance access from the network admission controller. In this mode diagnosis and maintenance traffic is prioritized to allow fast and efficient maintenance. If the request is granted, the admission controller instructs the network controller to configure the network by reserving the requested resources accordingly. Note that, after the maintenance access has been completed, the same principle can be used to release the network resources. This use case is also relevant during the initial flashing process at the OEM's assembly line.

Infotainment: A passenger wants requests to watch a video. This request is sent to the network admission controller, which decides if the additional video traffic can be supported by the network while sill providing sufficient independence to critical traffic streams. If the request is granted, the admission controller instructs the network controller to configure the network by reserving the requested resources accordingly. Note that, after the video has been watched, the same principle can be used to release the network resources.
Change of driving scenario: The driver initiates a change of driving scenario (e.g. highway to city). This requires, among other things, to switch the sensors of the vehicle's advanced driver assistance systems e.g. from unidirectional long range (highway) to omnidirectional short range (city). The mode change is requested from the admission controller, which then instructs the network controller to reserve the requested resources accordingly, i.e. based on the driving scenario, there might be significantly different traffic flows in the network. This mode change must finish in bounded time.
This use case will demonstrate the flexible network admission control and configuration capabilities developed during the project by means of the *workshop* scenario, as well as show their seamlessly functionality with the investigated network security methods.

5.3.5 Use Case 3.5: Attack Prevention

Actor	Attacker, e.g. malicious traffic stream
Goal	Prevent the attack from affecting the network
Trigger	Attack detection
Precondition	Network in normal operation
Postcondition	Attack is contained. There are no negative effects on other streams in the network.

Table 5.5: Overview Use Case 3.5: Attack Prevention

Description

Network attacks must be prevented from affecting critical traffic streams. Attacks can be detected by applying monitoring at the ports of network equipment, e.g. egress ports at end stations of ingress or egress ports of switches. Once an attack has been detected, it must be contained in order to prevent adverse behavior from propagating into the network. This can be done, for example, by either shaping or blocking, each of which can be done on a per-class or per-stream basis. It is part of WP5 to figure our which of these strategies is best suited for time- and safety-critical networks, e.g. whether detection and containment can be performed in bounded time. Note that this use case can also be applied to scenarios, which behave like network attacks, but have a non-attack related cause, e.g. babbling idiots flooding the network.
This use case will demonstrate network attack prevention/mitigation capabilities developed during the project. Various security aspects are also analysed. They pertain to confidentiality, authenticity and integrity, with the aim to prevent network attacks, whilst respecting real-time constraints. One way to protect the communication link from e.g., man-in-the-middle attack, is through the integration of cryptographic solutions. WP5 explores the adoption of minimal invasive methods for end-to-end encryption in a TTEthernet-based network. More precisely, METADAT Stream Cypher (MDSC) and MACsec (IEEE 802.1AE) are employed as cryptographic alternatives for making a network more secure against possible alterations in the communication link between two parties. The final results will allow to select the best performing method, which will be implemented to cover security aspects in the SAFURE framework. The timing effects of network attacks can be identified and quantified using a timing analysis approach.

5.4 Summary

One of SAFURE's goals is to propose methods and extensions to Ethernet that enable a safe and secure automotive communication infrastructure. The automotive network scenario defines a set of use cases which will be used to demonstrate and evaluate SAFURE's research results. Particularly, these use cases address the various safety and security aspects, which have been identified in the description of action. Table 5.6 gives an overview of the relation of the automotive network use cases to work packages and the SAFURE objectives.

Use Case	Type	Description	Work Packages	Objectives
3.1	timing	System-wide timing verification demonstrated across both automotive use cases	WP2, WP3, WP6	OBJ2
3.2	safety	Formal timing analysis of the impact of transient network faults	WP2, WP3, WP6	OBJ2
3.3	safety	Detection, recovery, and prevention of permanent network faults	WP5, WP6	OBJ2, OBJ3
3.4	safety	Admission control and isolation (sufficient independence) of mixed-critical network traffic	WP5, WP6	OBJ2, OBJ3
3.5	safety, security	Detection and prevention of different network attacks	WP5, WP6	OBJ1

Table 5.6: Use case relations to work packages and SAFURE objectives

Chapter 6

Conclusion

6.1 Summary of Scenarios

The three described scenarios are relevant for *security* and *safety* aspects in mixed-critical contexts. In particular:

- The **Telecom Scenario**, cf. Section 3, focuses on wireless connectivity between general-purpose smartphones and medical devices, and moreover, it is focused on smartphone mechanisms which separate the processing of business operation from other processes in order to guarantee a high assurance safety for health applications. This scenario is also applicable to the automotive domain, e.g. in the insurance sector.
- The **Automotive Multi-Core Scenario**, cf. Section 4, is focuses on anti-tuning, multi-core architecture, secure communication and mixed-critical contents for new generation ECUs.
- The **Automotive Network Scenario**, cf. Section 5, focuses on safety measures required to enable mixed-critical communication in future in-vehicle Ethernet networks. This scenario also focuses on security aspects of Ethernet, e.g. to prevent unauthorized parties from participating in vehicular communication.

All the described scenarios are an opportunity for the involved partners to go in depth and works together on various mixed-critical topics related to safety and security aspects. Table 6.1 gives an overview how the objectives stated in the SAFURE DoA will be covered by the use cases.

Use Case	**Type**	**Description**	**Work Packages**	**Objectives**
1.1	security	Improving security of BYOD and CYOD	WP2, WP3, WP4, WP5, WP6	OBJ1, OBJ2, OBJ3
1.2	safety, security	Body Area Network Cybersecurity	WP2, WP3, WP4, WP5, WP6	OBJ1, OBJ2
2.A1	safety, security	Modelling Safe & Secure software components in a multi-core PWT ECU	WP2, WP6	OBJ2, OBJ3
2.A2	timing	Timing Analysis of software component allocation on multi-core ECU	WP2, WP6	OBJ2
2.A3	safety	Data Protection on multi-core ECU	WP3, WP6	OBJ1
2.A4	security	Data Integrity on multi-core ECU	WP3, WP6	OBJ1
2.A5	safety	Mixed-critical ECU	WP4, WP6	OBJ1
2.B1	safety	ECU Normal Operation	WP6	OBJ1

Continued on next page

Table 6.1 – *Continued from previous page*

Use Case	Type	Description	Work Packages	Objectives
2.B2	safety	Safety ECU Faulty Operation	WP6	OBJ1
2.B3	security	Security ECU Faulty Operation	WP6	OBJ1
3.1	timing	System-wide timing verification demonstrated across both automotive use cases	WP2, WP3, WP6	OBJ2
3.2	safety	Formal timing analysis of the impact of transient network faults	WP2, WP3, WP6	OBJ2
3.3	safety	Detection, recovery, and prevention of permanent network faults	WP5, WP6	OBJ2, OBJ3
3.4	safety	Admission control and isolation (sufficient independence) of mixed-critical network traffic	WP5, WP6	OBJ2, OBJ3
3.5	safety, security	Detection and prevention of different network attacks	WP5, WP6	OBJ1

Table 6.1: Use case relations to work packages and SAFURE objectives

The industrial demonstrators, based on the three described scenarios, are provided by the telecommunication and automotive sector, and they represent a chance, for all SAFURE partners, to create new standards and to provide support for the start of new industrial products and new research projects.

6.2 Use of the scenarios and applications

All the themes arose in the examined use cases are of great interest for several embedded stakeholders, like: assurance market, medical sector, automotive OEMs, telecommunication market, and end users. All these stakeholders are always paying more attention to the safety and security aspects of their systems. All the described scenarios will be implemented on SAFURE demonstrators and all the solutions deployed can be partially or completely reused inside real industrial products. As an example, automotive solutions for safety and security will be partially reused for new generation Powertrain ECUs.

Bibliography

[1] AUTomotive Open System ARchitecture (AUTOSAR). Specification of Operating System. http://www.autosar.org/fileadmin/files/releases/4-0/software-architecture/system-services/standard/AUTOSAR_SWS_OS.pdf, November 2011. R4.0 Rev 3 V5.0.0.

[2] Wayne Burleson, Shane S Clark, Benjamin Ransford, and Kevin Fu. Design challenges for secure implantable medical devices. In *Proceedings of the 49th Annual Design Automation Conference*, pages 12–17. ACM, 2012.

[3] Freescale. MPC57xx MCUs. http://www.freescale.com/webapp/sps/site/taxonomy.jsp?nodeId=012FCB06C1EAD5&cof=0&am=0.

[4] Infineon. AURIX. http://www.infineon.com/cms/en/product/microcontroller/32-bit-tricore-tm-microcontroller/aurix-tm-family/channel.html?channel=db3a30433727a44301372b2eefbb48d9.

[5] International Organization for Standardization (ISO). ISO 26262: Road Vehicles – Functional Safety, 2011.

[6] Chunxiao Li, Anand Raghunathan, and Niraj K Jha. Hijacking an insulin pump: Security attacks and defenses for a diabetes therapy system. In *e-Health Networking Applications and Services (Healthcom), 2011 13th IEEE International Conference on*, pages 150–156. IEEE, 2011.

[7] Open Networking Foundation. OpenFlow. https://www.opennetworking.org/sdn-resources/openflow.

[8] Nathanael Paul, Tadayoshi Kohno, and David C Klonoff. A review of the security of insulin pump infusion systems. *Journal of diabetes science and technology*, 5(6):1557–1562, 2011.

[9] Michael Rushanan, Aviel D Rubin, Denis Foo Kune, and Colleen M Swanson. SoK: Security and privacy in implantable medical devices and body area networks. In *Security and Privacy (SP), 2014 IEEE Symposium on*, pages 524–539. IEEE, 2014.

[10] SAFURE - SAFety and secURity by design for interconnected mixed-critical cyber-physical systems. Annex 1 - Description of Action, 2014.

[11] Amy Tenderich. ADA Device Report: New 'Jewel Pump' is Best in Show. *Healthline*, 2010.

[12] THALES. TEOPAD: security for smartphones & tablets. https://www.thalesgroup.com/sites/default/files/asset/document/COM_WHITE%20PAPER%20TEOPAD_EN_V3.pdf.

[13] THALES. TEOREM - Fixed and cellular secured telephony solution for Governmental and Defence use. Leaflet, https://www.thalesgroup.com/sites/default/files/asset/document/teorem_leaflet_uk_29052008.pdf, 2008.

[14] U.S. Department of Health and Human Services - Food and Drug Administration (FDA). Content of Premarket Submissions for Management of Cybersecurity in Medical Devices, 2014.

[15] vector. ECU Analysis with CANalyzer. http://vector.com/vi_canalyzer_en.html.

[16] Wikipedia. Body area network. http://en.wikipedia.org/wiki/Body_area_network.