UNIVERSITY OF TARTU
Institute of Computer Science
Innovation and Technology Management Curriculum

Mariia Bakhtina

Securing Passenger's Data in Autonomous Vehicles

Master's Thesis (20 ECTS)

Supervisor: Raimundas Matulevičius, PhD
Co-supervisor: Mari Seeba, MSc

Tartu 2021

Securing Passenger's Data in Autonomous Vehicles

Abstract:

Autonomous vehicles (AV) are becoming a part of humans' everyday life. This thesis aims to determine how passenger's personal data can be protected in the autonomous vehicle. On the one hand, during the ride, autonomous vehicles are highly dependent on passenger's data usage, and the privacy of personal data is mandatory to be guaranteed to AV passengers. On the other hand, assuring the security in the Passenger–AV interaction is a required aspect to address, as along with opportunities, new cybersecurity risks and challenges occur.

Firstly, the thesis presents an approach of security risk management in the Passenger-AV interaction based on the ISSRM domain model. The research results in the identified protected assets and a threat model. The security risks are detected based on the proposed threat model, and corresponding security requirements are elicited. Secondly, the thesis demonstrates how the tool-supported business process analysis can be utilised for passenger's personal data privacy protection. We illustrate how tool-supported GDPR-compliance check can be exploited and how to use data disclosure analysis for preventing passenger's personal data leakage. Besides, the thesis presents a few designs proposing to adopt privacy-enhancing technologies for personal data protection.

The research is conducted in the lab settings in the form of a case study. The findings of the thesis are not dependant on the AV hardware architecture and can be generalised to other scenarios of Passenger–AV interaction. They are suitable for AV systems used by ride-hailing service providers that enable supervisory AV control. The presented data protection approach is also appropriate for other autonomous motor vehicle types that transport people.

Keywords:

Autonomous vehicles, information system security risk management (ISSRM), human-computer interaction, threat modelling, personal data protection.

CERCS:

T120 – Systems engineering, computer technology

Reisija andmete turve isejuhtivates sõidukites

Lühikokkuvõte:

Isejuhtivad sõidukid (AV) muutuvad osaks inimeste igapäevaelus. Siinse magistritöö eesmärk on tuvastada võimalusi reisija isikuandmete kaitsmiseks AV-s. Ühelt poolt AV-d sõltuvad sõidu ajal reisijate andmete kasutamisest ja isikuandmete privaatsuse tagamine peab seejuures olema AV kasutajatele tagatud. Teisalt on Reisija-AV vaheline turvaline suhtlemine oma uute võimaluste, küberturbe riskide ja väljakutsetega oluline teema, mida uurida.

Esmalt tutvustatakse magistritöös infoturvariski halduse käsitlusviisi Reisija-AV suhtluses, mis põhineb ISSRM domeeni mudelil. Uurimistulemusena tuvastati kaitstavad varad ja selle põhjal ohumudel. Turvariskid tuvastati väljapakutud ohumudeli alusel ja selgitati välja neile vastavad turvanõuded. Teiseks näitlikustatakse töös seda, kuidas reisijate isikuandmete kaitseks saab kasutada äriprotsessi analüüsi tööriistu. Töös kirjeldatakse, kuidas tööriista abil viia läbi GDPR-i nõuetele vastavuse hindamist ja kuidas kasutada andmete avalikustamise analüüsi, et vältida reisijate isikuandmete lekkimist. Lisaks tutvustab töö mõningaid projekteerimisvõimalusi, mis omakorda pakuvad välja privaatsust suurendavate tehnoloogiate kasutuselevõttu isikuandmete kaitseks.

Uurimustöö viidi läbi laboratoorsetes tingimustes juhtumianalüüsi vormis. Uurimuse tulemused ei sõltu AV riistvaraarhitektuurist ja neid on võimalik üldistada teistesse Reisija-AV suhtluse stsenaariumidesse. Tulemused sobituvad AV-süsteemidele, mida kasutavad sõidujagamisteenuse pakkujad ja mis võimaldavad AV-järelevalvet. Esitatud lähenemine andmekaitsele sobib ka teistele autonoomsetele sõidukitüüpidele, mis transpordivad inimesi.

Võtmesõnad:

Isesõitvad sõidukid, infosüsteemi turvariski haldus (ISSRM), inimese-arvuti interaktsioon, ohu modelleerimine, isikuandmete kaitse.

CERCS:

T20 - Süsteemitehnoloogia, arvutitehnoloogia

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

With the push of innovation, the automotive industry moves along the path of transformation to autonomous vehicles. An autonomous vehicle is one of the trending topics in the research community, supported by considerable investments from government, private firms, and research centers [1]. An autonomous vehicle (AV) is defined as a system that is able to conduct dynamic driving tasks with limited human intervention [2]. While creating a fully autonomous vehicle for everyday usage is still a big challenge, there are cars which has advanced self-driving features [3] and different projects are piloting autonomous buses [4, 5]. Therefore, it is just a matter of time when fully autonomous vehicles are integrated into humans' everyday lives.

With the development of self-driving cars, new opportunities for providing vehicle's passengers with value-adding services arise. Such services include information systems that give a passenger understanding of the vehicle's behaviour and surroundings (i.e. infotainment systems) and a possibility to engage in more advanced dialogue. Such a dialogue is referred to as *Passenger-AV interaction*, and it should enable a passenger to control the car behaviour, destination or selected route of the trip while AV can interpret high-level passenger's command by adjusting driving tasks accordingly. For enabling such supervisory control over the vehicle, the information systems (IS) that accompany the main driving functions are embedded in the AV. These systems collect and process data about passengers, their location, how they interact with the AV. While such data processing opens a range of new opportunities for user experience improvement, new security challenges arise. To trust self-driving technology and let it become ubiquitous, AV users need to be sure that their personal data is protected and can be accessed only by authorised entities with harmless intention. While local legislation regulates information privacy assurance, organisations are responsible for meeting legislative requirements and securing both users and organisational information. For example, misuse of AV systems may give an adversary access to sensitive data, and compromising the information system may threaten the passenger's safety [6]. Hence, the protection of the information needs to be assured by managing the corresponding privacy and security risks.

1.2 Scope

The increase of vehicle system automation leads to the concept switch from Human-Computer Interaction (HCI) to Human-Robot Interaction (HRI) as the nature of interaction design changes from control-oriented to supervisory control [7]. With the concept switch from HCI to HRI, new risks may occur, and therefore, it should be investigated whether the standard information systems security risk management (ISSRM) and data protection procedures are applicable to the Human-AV interaction. To do so, we research a Human-AV interaction to investigate whether the methods of ISSRM in HCI can be applied to it. The scope of the research is limited to a Passenger–AV interaction that is one specific scenario of Human-AV interaction (see Fig. 1). Such a specification enables more in-depth investigation from a passenger's perspective, excluding other humans that can interact with the driving AV (e.g., pedestrian, system administrator). Within the selected type of interaction, the information systems need to process sensitive personal data (e.g., geolocation of the passenger). Hence, the privacy of passenger's data should be assured. Data privacy directly depends on AV security as personal data is one of the assets controlled by the AV information system.

Therefore, to protect passenger's data in AV, one needs to address both privacy and security risks in the context of designed Human-AV interaction. Moreover, the need of ensuring information security is highlighted in the legislation [8, 9] that regulate data privacy (also referred to as 'data protection').

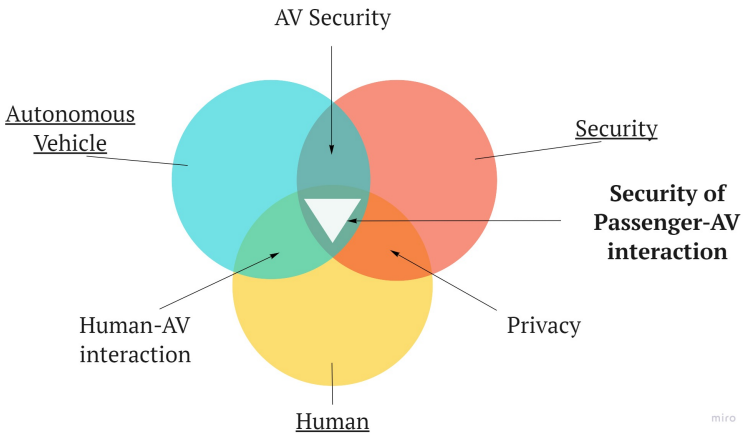

Figure 1. Context of Passenger-AV interaction security

So far, a few studies (e.g., [6, 10]) considered this interaction scenario from the perspective of information protection. A high-level impact of security attacks that threaten the infotainment systems which enable the Passenger–AV interaction is confirmed [11]. However, previous studies only highlighted the necessity of raising awareness of the passengers about possible security issues, not providing guidelines of how personal data should be protected and how corresponding risks need to be addressed. Thereby, there is a gap in the methods of protecting personal data within AV systems that is addressed in the thesis.

1.3 Problem Definition

While much attention is paid to making autonomous driving ready-to-use technology, the urgency of information security and data protection has not been studied with the same intensity [6, 10]. Security researchers are currently exploiting proof-of-concept attacks [6] to assess the quality of the produced cars with self-driving features. Numerous successfully implemented attacks proved the possibility of remotely taking control of a car's infotainment system and manipulating driving function. They demonstrate a lack of proper treatment of security risks during the system development. Security attacks on the autonomous vehicle may lead to vehicle damage, financial losses, disclosure of sensitive personal data [6]. The risks that may threaten passenger's data in the AV should be defined as fully as possible so that technology can be used in real-life settings. Therefore, an approach of personal data protection in the moving AV needs to be defined so that AV system owners would have a comprehensive guideline for assuring passenger's data privacy.

One way to address this task is to separate AV functionality into use cases and conduct a separate risk analysis. The developed AV system components and their use cases should be modelled from different perspectives. That allows conducting an in-depth analysis for defining the system vulnerabilities and potential threats. For instance, a Passenger–AV interaction case includes sharing data - personal and required transporting - between infotainment system, external ride-hailing service

provider, intelligent transportation system (ITS) components, e.g., AV controllers, and the other ITS infrastructure. Meanwhile, the communication between the AV system components that conduct dynamic driving tasks is characterised by other data flows and vulnerabilities which should be approached individually.

As an AV infotainment information system (IS) includes data flows that are not typical for researchers due to the transformation into more intelligent IS, there is a need for a systematic approach for data protection in the AV system. Currently, no such standardized way is defined neither for autonomous vehicles overall nor for a Passenger–AV interaction, specifically. This thesis proposes the approach of personal data protection in the selected interaction considering two concepts of its assurance - from the perspective of privacy management and the perspective of information security risk management. As a baseline for privacy management, we use local legislation requirements that establish rules of personal data processing and protect human rights of data privacy. For addressing security risks, we use a threat-driven approach which is widely used for guiding security risk management [12, 13, 14]. The investigation of security risks is conducted for the system that consists (i) of the in-vehicle systems which provides infotainment services to passengers and are in charge of dynamic driving tasks of a vehicle, and (ii) of a central IS that helps passengers to set interaction with the vehicle. It should be noted that the set of attacks on the in-vehicle network and to central IS are still present but are not considered as the main subject of the current investigation.

1.4 Research Questions

The following main research question is proposed:

How can the information in Passenger–Autonomous Vehicle interaction be protected?

Considering the usage of passenger's personal data in the researched interaction, it is the primary information to be protected. Privacy of personal data is defined by human rights that establish rules of personal data processing and is mandatory to be guaranteed to AV passengers. One of the principles required by data protection regulation is information security. The methods of handling personal data protection to preserve privacy differ from the methods of managing information security. Therefore, to address the main research question, we distinguish two approaches of assuring personal data protection - information security and general privacy management. So, the following questions should be answered:

RQ1: How can information security risks in Passenger-AV interaction be managed? Answering this question, we propose an approach for identification and addressing information security risks. Firstly, we identify the assets to be protected in the researched interaction. Then, we find security risks that may harm the defined assets. Finally, we define countermeasures for risks mitigation.

RQ2: How can the privacy of passenger's personal data be managed in the Passenger-AV interaction? After defining how to ensure information security in RQ1, we illustrate how to collect and process personal data lawfully by answering this question. We illustrate how tool-supported legislation compliance check can be exploited and how to use data disclosure analysis for preventing passenger's personal data leakage. The answer to RQ2 also presents a few designs of the researched business process, proposing adopting privacy-enhancing technologies for personal data protection.

The thesis aims to propose a baseline for systematically ensuring personal data protection in the business process of Passenger–AV interaction.

1.5 Contribution

This work contributes to existing knowledge of autonomous vehicle systems management by providing the following results:

1. The thesis introduces a threat model for the AV systems that helps to apply a threat-driven approach and can be used as a foundation in managing information security risks in Passenger–AV interaction. The proposed model's value is illustrated by its use as a baseline for the risks deriving and requirements elicitation during the case study. Hence, an information security specialist and the Passenger-AV interaction process owner will benefit from the provided threat model using it as a baseline for their security systems analysis procedures in AV system development or support organisations.

2. The thesis contributes to the AV system management process by illustrating the utility of tool-supported privacy analysis. Thereby, a data protection officer will benefit from the thesis results by understanding how the organisation can benefit from using specialised tools for privacy analysis.

3. Finally, we propose a set of security measures for the data protection within the Passenger–AV interaction scenario. Thus, AV system developers will benefit from the presented security control mechanisms by using them as possible ways of meeting security requirements during the AV system design and development.

1.6 Structure

The rest of the thesis is structured as follows. **Chapter 2** contains an overview of information security risk management and personal data management procedures. The differences in assuring information security and privacy of personal data are defined. The methods to be used in this work are discussed. This chapter also describes the case design used in the thesis. The research method for addressing personal data protection in the selected case scenario is described. **Chapter 3** describes how information security risks can be managed for the case scenario. The protected assets, security risks, and possible countermeasure are presented. **Chapter 4** demonstrates the results of tool-supported case analysis on passenger's data protection. The chapter begins by analysing the case business process's compliance with the GDPR requirements and further analysis of privacy-enhancing technologies usage in the context of the case scenario. Additionally, a few designs of personal data protection are proposed. Finally, **Chapter 5** concludes the thesis by providing answers to the research questions, the limitations, conclusions, and defining vectors for future work.

2 Background

Assuring data protection is an important part of the system development lifecycle, which aims to support the reliability of a developed system and protect privacy right for the processing of personal data. This chapter gives a literature review, examining personal data protection approaches for ensuring data privacy. We discuss data protection principles, including assurance of information security and usage of privacy-enhancing technologies, method of data privacy management (e.g., usage of tool-supported privacy analysis, security risk management). The chapter also includes an overview of autonomous vehicles information system components and discusses how information security has been previously considered for the AV systems.

2.1 Personal Data Management

Personal data is a particular type of information. In general, the protection of personal data is addressed according to the guidance of information security and security risk management [15] as it is discussed in the next sub-section. However, the nature of personal data requires to treat its protection in a more specific way, namely, by national legislation. Therefore, this subsection provides the reader with the necessary background about how personal data managed on the legislation level in the European Union (EU) by covering the crucial aspects of the General Data Protection Regulation [8]. Here we also introduce the main concepts that support the regulation, namely privacy by design concept, privacy-enhancing technologies (PETs), and tools used to verify data protection compliance with the legislation. Additionally, we discuss how personal data privacy is treated in some countries outside the EU by comparing their local legislation with GDPR.

2.1.1 Privacy and Personal Data Protection

During the last decade, there is an increase in companies that leverage information and communication technologies like networks and digital solutions to provide services to their customers, claims ENISA in [15]. Thus, companies rely on a high volume of information that is processed and needs to be secured. While information security risk management focuses on managing any valuable for the business assets, there is one specific information type that organisations should treat in a specific manner - *personal data*.

According to [8, 16], *personal data* refers to the information about an identifiable person and, therefore, tightly connected with the human rights and internal national regulations. The concept of "*privacy*", however, does not have one specific definition. According to [17], privacy is a multidimensional concept. One of the aspects is the privacy of personal data, which is the main subject of protection in the context of information security. *Privacy of personal data* is defined by human rights and establishes the rules of personal data processing. In essence, data privacy specifies who is authorised to access personal data and how it can be processed. Meanwhile, *data protection* regulates in details different instances of personal data processing through the development of key legal principles [17]. Thus, while the concepts of *privacy* and *data protection* are interconnected, they have different focuses. The latter is concentrated on personal data, while the former is focused on the individual per se.

For assuring the privacy of personal data, nations formulate guides of personal data protection in the form of national laws. Hence, here we present a review of the law that regulates personal data

privacy within the European Union. Additionally, we compare the EU legislation with laws in two other regions to understand the main used concepts the organisation should pay attention to when processing personal data.

European Union. Since 25 May 2018, in the European Union (EU), a new personal data regulation framework has gained its power - the **General Data Protection Regulation** (EU) 2016/679 (i.e. GDPR) [8]. GDPR is a regulation that means it has binding legal force throughout every Member State [18], thereby harmonising all the existing legal frameworks across the Member States regarding data protection.

According to GDPR [8], there are a set of definitions that should be introduced to understand the legislation. *Personal data* is "any information relating to an identified or identifiable natural person ('*data subject*') [...] such as a name, an identification number, location data". The company or other body which "determines the purposes and means of the processing of personal data" is called a *controller*, while *processing* refers to "any operation [...] which is performed on personal data." Another company, which provides services and processes personal data on behalf of the controller, is called a *processor*. Another important term to consider is *consent* of the data subject which refers to any "freely given, specific, informed and unambiguous indication of the data subject's wishes by which he or she [...] signifies agreement to the processing of personal data relating to him or her".

One of the main requirements to data controllers and processors stated in GDPR (Section 2) is ensuring the *security of personal data*. It includes keeping "confidentiality, integrity, availability and resilience of processing systems and services" by implementing appropriate technical (*security measures*) and organisational (*data protection policy*, also known as *privacy policy*) measures. Moreover, each controller and processor should maintain a *record of processing* activities. One should pay attention that within GDPR, there is no usage of the term "*privacy*" and the term "*data protection*" is used as a synonym to the former when talking about assurance of information privacy.

Currently, most of the data processing is digitalised and is conducted by processing systems, and thus, ENISA claims [15] that risks for personal data privacy are primarily caused by security risks of the IT applications and networks. Therefore, for each company that handle personal data inside the EU, there is a need to address personal data management in compliance with GDPR. One of the steps in this task resolving is examining information systems used for data processin and privacy policies to mitigate security risks.

Singapore. In 2012 the **Personal Data Protection Act 2012** (No. 26 of 2012) (PDPA) [16] came into effect in Singapore. The act regulates personal data protection for private organisations, excluding public agencies and organisations which act on behalf of public agencies.

In general, according to [19], PDPA is a similar to GDPR comprehensive law, which covers an analogue scope of personal data collection, use and disclosure but has some differences in the used terminology comparing to the EU legislation. Among other differences of personal data protection, the next ones are the most important to consider while comparing Singapore and EU regulations: (*i*) there is no differentiation of personal data special categories; (*ii*) there are no mandatory requirements for organisations to keep a record of personal data processing activities; (*iii*) currently there is no mandatory requirements for organisations to notify supervisory authorities or affected individuals about data breaches; (*iv*) a data subject does not have the right to request the erasure of their personal data.

California, United States. While the United States of America does not have a comprehensive federal privacy law [20], **California Consumer Privacy Act** (CCPA) of 2018 is a state statute that introduces privacy rights and consumer protection for residents of California, Unites States. Since January 2020, CCPA regulates the collection, usage, sharing and selling of personal data. Among the main aspects which vary in CCPA comparing to GDRP are the following [9]: (*i*) the law obligations apply not to all organisations that operate in the selected territory, but only to those which meet either minimum threshold of the annual gross revenue or if a significant amount of revenue depends on the personal data sell; (*ii*) the scope does not cover such categories of personal data as employee data, medical information, personal information under the Driver's Privacy Protection Act; (*iii*) the right to opt-out is only available for the cases of selling and sharing personal data and is not applicable for the data collection; (*iv*) no legal grounds are required for collecting and selling personal data on their basis; (*v*) privacy policy shall be updated at least every 12 months. To amend CCPA in November 2020 **California Privacy Rights Act** of 2020 (CPRA) was presented. CPRA expends the right of a consumer to limit usage and disclosure of sensitive personal information, including precise geolocation, introduces the right to rectify personal data, and obligates businesses to implement data protection by design and by default as well as to maintain a record of processing activities. Thereby, CPRA that comes into force in January 2023 will align CCPA more closely to the European GDPR.

To sum up, personal data protection regulations throughout the world have several similarities in scope, terminology, personal rights, and business obligations. Also, in different regions, the law regulating data privacy can be called differently. Most legislation that protects privacy in general also covers information privacy. Therefore *data protection* and *privacy* in the context of personal data can be used as interchangeable terms. The differences should be considered when running the business in different countries, as personal data protection rules may vary. Despite that, the organisation that manipulates personal data should define the privacy policy that guides personal data usage with respect to the local data protection or privacy law.

2.1.2 Privacy Principles and Privacy-Enhancing Technologies

Considering the data protection regulations, the data protection principles have become an integral part of any information system. Primarily, it is defined in GDPR (Art.47) that "the application of the general data protection principles, in particular purpose limitation, data minimisation, limited storage periods, data quality, data protection by design and by default, the legal basis for processing, processing of special categories of personal data, measures to ensure data security." *Privacy by Design* (PbD) is a principle of system design that aims to improve overall IS data protection [21]. The main rule of PbD is addressing privacy requirements throughout the whole system development lifecycle. It means that already starting from the system design, the privacy aspects should be considered and the design strategies and patterns should be used.

Hoepman at the [21] says that while *design strategies* are applicable in the phases of the system concept development and analysis, *design patterns* supports the system design phase. Later on, at the stage of system development to address data protection, privacy-enhancing technologies should be implemented [22]. Overall, there are eight data privacy strategies [21, 22]: minimise, separate, aggregate, hide, inform, control, enforce and demonstrate.

Different privacy-enhancing technologies are used to follow the mentioned privacy strategies. According to [22, 23] the term *privacy-enhancing technologies* (PETs) embraces the technical measures which ensure the privacy of personal data without the loss to the main IS functionality.

Pullonen et al. [23] divided PETs into five classes - *communication protection, data protection, entity authentication, privacy-aware computation*, and *human-data interaction*. The classification arranges technologies based on their application goals that can be achieved with the defined targets. For instance, the met *integrity* and *confidentiality* targets enable to achieve the *data protection* goal. In the same paper Pullonen et al. have introduced PE-BPMN – an extension of BPMN 2.0 language syntax to support PETs. The extension covers the technologies' classes of goals and targets by introducing *general stereotypes* (e.g., ProtectConfidentiality, OpenConfidentiality for targeting confidentiality) and *concrete stereotypes* that describes actual PETs (e.g., PK encryption, PK decryption for targeting confidentiality). PE-BPMN extension, thoroughly described in [24], helps a system analyst capture the PETs usage within the developed system.

PETs are mostly represented by cryptographic algorithms that employ mathematical techniques for addressing information security goals. Cryptography techniques assure secure data communication between a sender and receiver in the presence of an adversary. The central concept of cryptography [25] is *encryption*. The *encryption* refers to the transformation of a original data M (i.e. *plaintext*) to a protected data C (i.e. *ciphertext*) which differs from M. In order to transform the *ciphertext* C back to the *plaintext* M, a *decryption* transformation is applied to the ciphertext. The encryption and decryption transformations use *keys* as a mandatory input parameter for each transformation mechanism. Thus, a message sender and receiver need to know a key for conducting encryption and decryption transformations, respectively. If an adversary obtains access to the encrypted transmitted data, the original data is not disclosed as the representation of transferred data differs from the original, and for restoring it, the correspondent key is needed. As a result, in the case of the transferred encrypted data [25], a channel need not be secure as even gaining access to the transmitted data, an adversary cannot read ciphertext without knowing the exact decryption transformation. Thereby, encryption is often used for achieving confidentiality of the sensitive data that should be transmitted via the channel.

The encryption schemes differ one from another by the characteristics of applied transformation. In the context of personal data management, the fully homomorphic encryption transformation is the one that worth the attention during choosing PETs. The *fully homomorphic encryption* (FHE) [26] is encryption that supports arbitrary computation on ciphertexts. Thus, FHE enables to conduct processing of the encrypted data without its prior decryption. One application of the FHE is applying it to the personal data stored on the untrusted server in the protected form. The untrusted server can process the data, for example, as anonymous data for the algorithms or execute user's data queries without having access to the sensitive data itself. As a result, protected by FHE, data can be stored or processed by the untrusted party that provides the data owner with cloud services. There are several fully homomorphic encryption schemes; for example, *public key encryption* or *secret sharing* may implement FHE.

2.1.3 Privacy Analysis Tools

To assess the delivered privacy within the designed business process and information systems, the manual analysis can be conducted, or specialised tools can be applied. There are many commercial GDPR compliance checker and privacy impact assessment tools found on the Internet (e.g., PIA software by CNIL[1], secureprivacy.ai[2], etc.). Furthermore, specialised businesses provide services of

[1] `https://www.cnil.fr/en/`

[2] `https://secureprivacy.ai/`

checking the compliance of web solutions. However, the before-mentioned tools conduct the check of a deployed web application. Thereby, they are not applicable for the information systems in the early stages of software development or which are not a single web-based application, and therefore, cannot be used for the case analysis in our research.

Meanwhile, European Union's Horizon 2020 research and innovation programme[3] supports projects aiming to deliver solutions for data controllers and processors to help them meet the GDPR requirements. SPECIAL[4] and SPARTA[5] are among such research project.

In [27] authors demonstrate results of the SPECIAL project, which aims to provide a set of tools for data controllers and processors to check compliance with the GDPR automatically. The project resulted in the approach for the legislation compliance check and its initial performance evaluation based on the prototypes. However, the prototypes were case-specific and cannot be used for the researched scenario.

SPARTA is an ongoing project that conducts advanced research in cybersecurity. One of the project directions is security by design framework for the intelligent infrastructure, which also aims to address the EU data protection regulation. So far, the project delivered a prototype tool for checking the GDPR compliance - DPO tool.

This subsection reviews two tools that help to analyse a business process from the data protection perspective. The first tool - the DPO tool - can be used for primary analysis of the business process to define its compliance to GDPR, and the second one - Pleak - for analysing the quality of selected PETs for assuring prevention of personal data leakages.

DPO Tool DPO Tool[6] is a prototype which implements a model-based approach for achieving GDPR compliance of business process. The tool evaluates the compliance of a business process to the GDPR by comparing a BPMN model of the assessed process and the GDPR model.

The method of achieving GDPR compliance with DPO tool is presented in [28] and is based on the conceptual GDPR model, refined in [29] and [30]. The GDPR model defines the main concepts of the legislation that should be considered in the business processes and is depicted in a class diagram.

The following steps should be performed to conduct a tool-supported analysis using the DPO tool. Firstly, the As-Is business process model for the compliance check should be derived and uploaded to the tool. At this step, a user should also specify the main GDPR model elements (e.g., data subject and personal data, controller and processor, processing activities and technical measures). In the next step, the DPO tool compares the provided As-Is model with the GDPR model and gives the report in the form of instantiated GDPR model highlighting by flags non-compliance issues. If there are defined non-compliance issues, on the next step, the process owner should change the business process model by resolving non-compliance issues that finalises the regulation compliance check. Finally, the changed business process model can be used as a new As-Is model for the next iteration of the compliance check process.

The DPO tool has several advantages. First of all, it is an open-source prototype to be freely accessed and used by any process owner. Secondly, the tool can be used for the GDPR compliance check of a business process during the whole software lifecycle - at the design, development, or

[3] https://ec.europa.eu/programmes/horizon2020/

[4] https://www.specialprivacy.eu/

[5] https://www.sparta.eu/

[6] DPO Tool can be accessed at https://dpotool.cs.ut.ee/

operation phase - as it requires only the BPMN model for the input and not the working deployed application (that is in contrast required by the commercial solutions). Finally, in [31] comparison of the manual regulation compliance achievement using the method with the tool-supported analysis was presented. The results revealed significant correspondence of the identified non-compliance issues and pointed to the found non-compliance issues using the tool, which were omitted during the manual check.

On the other hand, the tool embraces several disadvantages. DPO inherits limitations of the used model-based approach presented in [32]. Firstly, it does not consider the national adaptations of GDPR by the EU Member States. Secondly, not all the GDPR requirements can be modelled, and, therefore, the model concerns 40 articles out of 90.

To sum up, the DPO tool is a proof-of-concept prototype that can be used as an alternative for the manual GDPR compliance check of business processes, keeping in mind the existing limitations of the used approach.

Pleak Tool Pleak (Privacy Leakage Analysis)[7] is an open-source web application prototype for analysing business processes and detecting possible data leakages [33]. The tool helps to investigate business process, which includes collaborative data processing and communication by providing a clear understanding about what data is leaked, on which extend and to who. Pleak enables to analyse the effect of used privacy-enhancing technologies in the context of the business process.

The tool aims to help with risk analysis and impact assessment of data privacy within the existing system. Pleak primarily targets non-information security specialists, like software developers or process owners, who need to assess the effectiveness of the selected privacy-enhancing technologies.

As an input, the tool gets the PE-BPMN model that depicts the analysed system through business actors and processes, data objects, and data analysis algorithms. As a result, Pleak provides the user with analysis reports of different depth levels. Overall, the report contains information about private data flows through the system, reveals actors to which the data is disclosed and to which extent.

The major advantage of Pleak is that the tool is multi-functional. Pleak proposes several techniques of the privacy analysis besides the described earlier [33], namely, leaks-when analysis, sensitivity analysis and differential privacy, guessing advantage analysis. The tool has an extensive tutorial, and Pleak verifies the provided model's syntax that makes the solution easy to use without previous expertise in privacy analysis or PE-BPMN.

At [23] the limitations of the tool are pointed out. In the current version of Pleak, the whole process is expected to be described in one diagram so that data objects are initially publicly accessible and then protected or processed. As the tool is under development, it currently supports the limited combinations of PE-BPMN stereotypes that restrains the analysis of privacy-enhancing technologies.

All in all, the tool can be used to analyse the currently supported PETs and define the system's vulnerabilities. Alternatively, Pleak can help compare PETs from one class in the context of the developed system for selecting the technology that enables better personal data protection.

[7] Pleak Tool can be accessed at `https://pleak.io/` (account: *demo@example.com*, password: *pleakdemo*, manual: `https://pleak.io/wiki/`) [33]

2.2 Security Risk Management

According to [34], *security risk management* (SRM) is defined as “an analytical procedure that helps us identify system valuable assets, stakeholders and operations. [...] It also provides logic and guidance to find and implement appropriate solutions for specific situations and mitigation strategies.” *Information security* is defined in [35] as a state which organisation assures "to protect the confidentiality, integrity, and availability of information assets, whether in storage, processing, or transmission" against adversaries. For controlling information security, organisations should manage security risks within the used information systems responsible for information assets. Thereby, information security risk management (ISRM) refers to managing security risks of organisational information in general, while information systems security risk management (ISSRM) refers to handling security risks that target information assets within the organisation’s information system. In the thesis, we focus on ISSRM, aiming to define how information can be secured utilising an information system and not focusing on protecting information once it is out of the organisation’s information system.

There are a number of *security risk management standards* which guide the activities for managing security risks in information systems, for example, ISO/IEC 2700x series [36, 37, 38], NIST special publications [39, 40]. In general, the introduced methods, perspectives, and terminologies of security risk management vary from one standard to another. The ISSRM domain model [41] combines principles of the standards mentioned above that enable us to generalise information system security risk management. According to the results of a systematic literature review in [42], the ISSRM domain model was assessed as one of the most proficient concepts that implement ISO/IEC 27001 standard requirements. Depending on the risk analysis approach (quantitative or qualitative), the nature of the problem, and the analyst preferences, organisations are employing different security risk management methods [43, 34] like OCTAVE [44], the NIST Cybersecurity Framework [45], MEHARI [46]. Nevertheless, to analyse the defined case, which is focused on determining security risks within the information system, we are using the ISSRM domain model as a baseline for the current research of SRM. The domain model is supported by security modelling languages [43] (e.g., SecureUML, security extension to Business Process Model and Notation (BPMN) language), which helps cover the model’s concepts using the corresponding tools.

According to the ISSRM domain model [41], there are three key concepts groups that should be considered - *asset*-related, *risk*-related, and *risk treatment*-related concepts. The model in the form of a UML class diagram is represented in Figure 2.

The *asset*-related concepts describe the organisation’s assets and their value to the organisation. The *business asset* represents any type of information vital for the proper processes flows and for achieving business needs. The *(IS) assets* support the defined business assets and are responsible for generating, manipulating, and storing new business assets. Mostly, IS assets are represented by information system components, such as software, network or hardware. The *security criteria* of a business asset are defined by security objectives, which describe the security need of a system. The CIA (Confidentiality, Integrity, Availability) triad forms the main security criteria that can characterise business asset and should be delivered by the corresponding IS assets.

The next group contains *risk*-related concepts, which correspond to risk itself and its constituent components. According to the ISSRM model [41], *risk* is composed of a *threat* that exploits IS asset’s one or few *vulnerabilities*, that leads to a harmful *impact* on the *assets* by negating the business asset’s *security criteria*. In turn, a *threat* is an incident that is defined by a *threat agent*

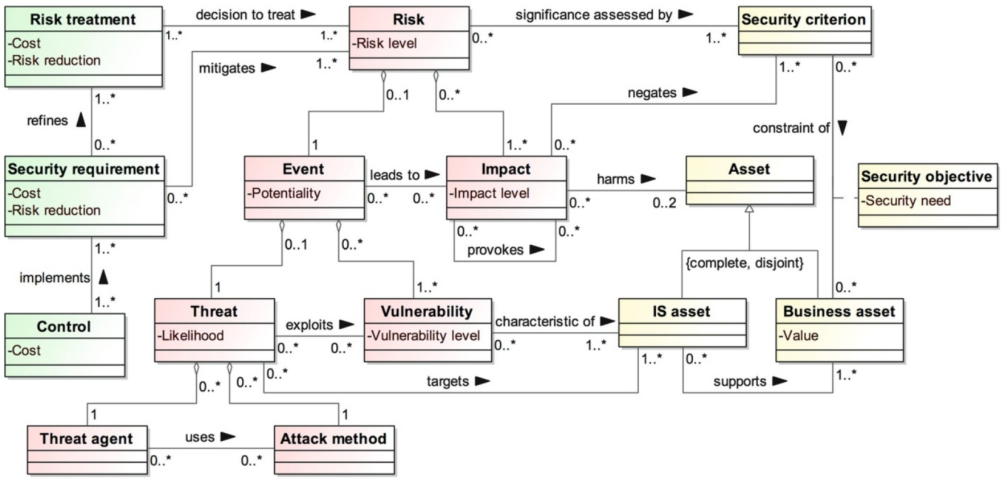

Figure 2. The ISSRM domain model [34, 41]: yellow entities represent asset-related concepts, red entities - risk-related concepts, and green entities - treatment-related concepts

(also referred to as an adversary) who uses an *attack method* intending to target the selected IS asset. It should be noted that using the ISSRM model we are focusing only on risks which are related to the intentional harm to system components caused by the malicious activity (i.e., attack method) of a threat agent.

Finally, *risk treatment*-related concepts describe decision regarding risk treatment. *Risk treatment* is refined by the *security requirements* that mitigate one or more risks and which are implemented by means of *controls*. Meanwhile, the decision about risk treatment is made based on the risk level (composed of risks likelihood and impact) and the related costs. There are four categories of risks treatment strategies [34]: avoidance, reduction, transfer, and retention.

To sum up, for conducting security risk management based on the ISSRM domain model, firstly, the protected assets should be defined for the researched IS, then, risks and finally, possible treatments. Conduction security risk management for the information systems enables organisations to be aware of the possible risks that may harm organisational assets and establish a risks management plan.

After reviewing the background of information security and personal data privacy, we conclude that both concepts guide personal data protection but using different approaches. The main difference between them can be formulated as follows. Information security risks can be accepted if the likelihood of the risk is low; the decision to accept the risk to personal data privacy is always wrong as it has an impact on individual and is regulated by the law. Therefore, in terms of the required action, the security risk management introduces recommendations that 'Should' be implemented, but, in contrast, personal data protection introduces principles that 'Must' be implemented. Furthermore, information security is one of the principles of assuring personal data protection and is still mandatory for managing. However, the level of delivered information security varies from organisation to organisation, while data protection should be provided on a similar level within all the organisations according to the local legislation guides.

2.3 Autonomous Vehicles

During the last few decades, the concepts of autonomy and autonomous systems have progressed considerably [3]. With the development of processors, sensors and actuators, navigation and communication [3], it became possible to consider the fifth level of automation that corresponds to a fully autonomous system. Thus, after the successful implementation of such partially autonomous systems as swarms, unmanned vehicles and the Internet of Things (IoT), researchers all over the world are trying to bring autonomy to the real world by prototyping autonomous vehicles [3]. Already nowadays, self-driving technology has been adopted by Tesla, Uber, Waymo [3], while many projects throughout Europe are piloting autonomous buses [4] for the usage of regular citizens.

An *autonomous vehicle* (AV) is defined as a vehicle with a system that can conduct dynamic driving tasks with limited human intervention [2]. Considering the literature review conducted by Affia et al. [47] and the vehicle system model from IPA [10], Figure 3 represents the AV System model.

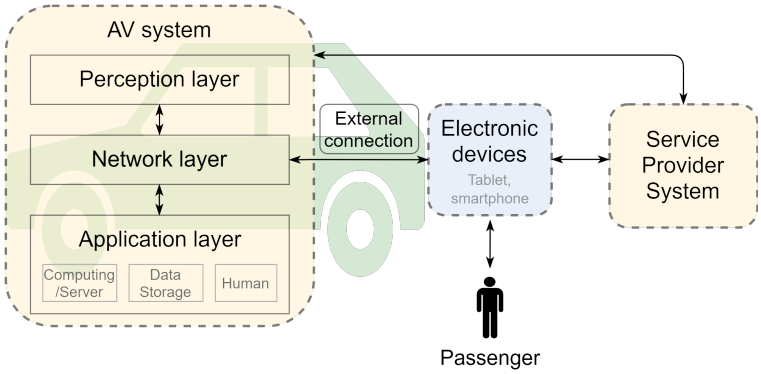

Figure 3. The autonomous vehicle system model (adapted from [10] and [47])

As it was described by Affia et al. [47], the AV system consists of three main layers.

- *Perception layer* is represented by hardware and software for sensing, positioning, seeing and managing data from the environment.

- *Network layer* enables transmission of data in-vehicle and outside (vehicle-to-vehicle (V2V) communication and vehicle-to-infrastructure (V2I) communication).

- *Application layer* connects the network layer with computing, storage and high-level processing of the generated and transmitted data. It includes control-relevant functions that are performed by computing and operating data storage. The application layer also enables passengers' comfort and user-friendliness by providing information and entertainment (also referred to as infotainment) [10]. Such an infotainment system allows a passenger to obtain information about the vehicle's current status, surrounding and allow to conduct strategic decisions (e.g., changing the destination of the ride). The infotainment systems can be either provided by vehicle manufacture or external service provider thanks to an electronic device (e.g., tablet or smartphone) in the car, communicating with the AV system.

In this master thesis, we examine how information security risks in Passenger-AV interaction can be managed (RQ1), focusing on the system's network and application layer. Moreover, we examine how the owner of information systems used for the interaction scenario should manage the privacy of passenger's data (RQ2).

2.4 Related Work

Numerous studies have attempted to address information security and risk management in autonomous vehicles looking at the problem from different perspectives. Most of the researches, like [48], [49] and [50], are focusing on the security of in-vehicle components, vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication. While AV-Human interaction mostly considers pedestrians, the role of passenger is not widely discussed as well as the role of external service providers (e.g., ride-hailing service or ride-sharing) in the interaction. Thus, in this study we limit the vehicle's interaction perspective by addressing a passenger directly and actively interacting with the AV system.

El-Rewini et al. [49] discussed vehicles the AutoVSCC (autonomous vehicular sensing-communication-control) framework and its role in vehicular communication. Additionally, they surveyed on vehicular components' vulnerabilities and security threats on in-vehicle and vehicle-to-anything (V2X) communications supplementing with potential countermeasures. As autonomous vehicles are supposed to become a part of the intelligent transportation system (ITS), the security aspects of V2X interaction in the research community are widely discussed in the context of VANET. As such, we use this paper as the supportive source for defining security challenges in the researched scenario which illustrates the particular case of the discussed in the paper V2X interaction.

Mejri et al. [51] conducted the survey on security challenges in VANET and identified security problems in the vehicle network, and provided corresponding cryptographic solutions. In [50], the authors discussed security and privacy threats in the case of autonomous and cooperative automated vehicles covering in-vehicle, V2I, and V2V interaction. The study highlighted some attacks which can take place in Passenger–AV interaction, such as attacks targeting in-vehicle devices (represented by hand-held devices connected to the vehicle's infotainment system via USB, Wi-Fi or Bluetooth), any other kinds of electronic devices of a passenger and maps (used by a vehicle in case of non-real-time detection of the road). Another essential aspect that endangers a vehicle's security is the tight connection of AV with hand-held devices. Mainly, the authors reviewed the users' personal devices that help viruses and malware invade into the vehicle's electronics through the infotainment system and harm the in-vehicle network and its components functionality. We can see that considering autonomous vehicle security there are various attack vectors which researchers are addressing. With respect to it and the increase of AV technologies development, Thing and Wu in [48] proposed a comprehensive taxonomy of AV attacks and defences that assist AV system architectures development. In [52] authors emphasised the increase of potential risks that affect or are conducted by the vehicle passengers as they have direct physical access to the system. Moreover, the research identified the following knowledge gap: "it is unclear what personal data will be generated and stored, [....] who owns it, and what potential risks there are." Thus, our work is complementary to the studies in [52, 48, 51] as we study the scenario of Passenger-AV interaction covering security attacks which target vehicular network, specific vehicle system component and passenger's device. As a result, the current work embraces the scope of the mentioned works by illustrating their findings on a particular scenario.

Concerning the security risk management of AV systems, there exist several comprehensive guidelines that should be mentioned. ENISA project [6] considers the passengers of AV only in the context of how different attacks threaten the passenger's safety and pinpoints a need of raising awareness of passengers "with respect to security issues and how to prevent them, on a regular basis." In contrast, provided by IPA guide [10] overviews the potential threats to autonomous

vehicles on the high level of abstraction and gives the general recommendations regarding security efforts in phases of automotive systems' lifecycle, which are not scenario oriented, but rather system functionality focused. Moreover, they consider a passenger as a passive system user, which only obtains information from the infotainment system. In [11] authors defined in-vehicle infotainment systems as the one that presents the biggest attack potential for vehicle networks. Additionally, the mitigation techniques and procurement recommendations for infotainment systems which enables passengers'/drivers' interaction with a vehicle was presented. This thesis aims to build an analogue guideline for the AV system developers and service providers that use AV systems for active Passenger–AV interaction.

To sum up, previous works discussed either general attack vectors on autonomous vehicle systems and defences rather than examples of their concrete applicability for the systems with a specific architecture or the general guides of the vehicular system's security risk management. While the significant harm of the attacks on the infotainment systems is highlighted, none of the studies comprises a comprehensive overview of managing information security and privacy. Therefore, this thesis aims to illustrate the systematic approach of ensuring data privacy and data protection within autonomous vehicle systems. The presented results are supposed to be used by security specialists, software developers, Passenger–AV interaction process owners, and data protection officers in AV development and service provider companies during a system development lifecycle (SDLC) starting from the stage of the system design.

2.5 Case Design

In this thesis, we conduct a case study of the Passenger–AV interaction scenario. The case is based on the business process and the AV system architecture designed in the autonomous driving lab[8] of the Institute of Computer Science, University of Tartu. The research is primarily conducted in collaboration with a ride-hailing company - Bolt Technology OÜ (further referred to as 'Bolt'). Therefore, the designed ecosystem is supposed to be used by a ride-hailing service provider to allow customers to use driverless ride-hailing services.

The Passenger–AV interaction is a part of the Ride Fulfilment process, which consists of three parts that deliver value to a Passenger, namely *Ride Initiation*, *Ride Execution* and *Ride Post-Processing* (see Fig. 4).

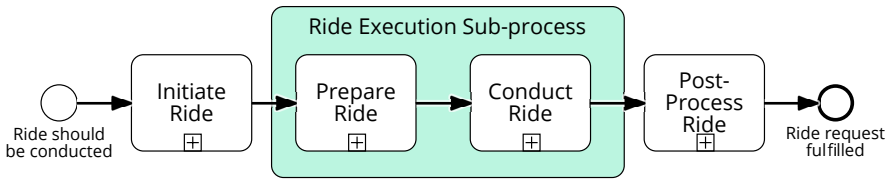

Figure 4. Value Chain of Ride Fulfilment Process

As the baseline of the surveyed processes, we used a user interface prototype [53] designed in the autonomous driving lab. The prototype was designed aiming to increase trust in autonomous vehicles. For further case analysis, we have depicted the Ride Fulfilment business process using

[8]More information about the autonomous driving lab can be found at https://www.cs.ut.ee/en/autonomous-driving-lab

BPMN language. The process model is presented in Figure 5, and the more detailed version of this process enriched with the considered in the thesis data artefacts and activities can be found in Appendix II.

Ride Initiation The process starts when a *Passenger* creates an account filling in information about themselves into the *Profile Data* artefact. Having a profile, the Passenger requests a ride by submitting *Ride Details* in the Service Provider's mobile app accessible via a personal device. To approve the received ride request, *Central System* creates *Ride Details* based on the provided by *Passenger* artefact, assigns a vehicle to the ride and sends the copy of *Ride Details* to *AV System* of the assigned vehicle to execute the ride. Receiving details of the assigned ride, the AV System processes it and gets to the starting point of the ride.

Ride Execution Approaching the starting point, *AV System* generates *Device Notifications* and sends it to *Central System*. *Central System* creates *Passenger Notifications* based on the received *Device Notifications* to be sent to Passenger via the mobile app. Entering the vehicle, *Passenger* should use installed on the backseat tablet where the web client of *AV System* is displayed. To establish interaction with the vehicle, Passenger should use provided in the *Passenger Notification* token code to authenticate themselves to *AV System* by entering the authentication details. After, *Passenger* can choose a ride route among the presented and initiate the start of the ride. In the move, *AV System* uses *Ride Details* as a baseline for controlling the ride and conducting dynamic-driving tasks.

Meanwhile, *AV System* generates *Passenger Notification* to be shown to Passenger in order to inform them about the current location on the map, current speed, and driving mode. In turn, *Passenger* can create *Ride Update Request* to manage the ride - either change the destination or change the driving style - which *AV System* processes to adjust the *Ride Details* respectively. Additionally, *Ride Update Requests* are stored to the storage for justifying the changes of initial Ride Details and for further storage and algorithms adjustments. Approaching the destination, *Passenger* can select a spot to get off among the presented, and after reaching the destination, *Passenger* is asked to leave the vehicle. On this point Passenger–AV interaction is finished.

The Ride Execution sub-process can be logically separated into two big tasks – *Ride Preparations* and *Ride Conduction*. *Ride Preparation* refers to establishing Passenger–Vehicle interaction with the support of the *Central System* and conducting other settings before the ride starts. *Ride Conduction* includes the continuing managing of the ride by *Passenger* and continuing controlling the ride and reacting to Passenger's commands by *AV System*.

Ride Post-Processing After Passenger left the vehicle, *AV System* should transfer *Ride Update Request Storage* copy to *Central System*. *Central System* finalises the ride by analysing the captured during the ride data to improve future services and archive Ride Detail to the storage to access them on demand.

Additionally, the architecture of the researched system and its context should be noted. As shown in Figure 3, the considered ITS incorporates (*i*) AV system, which is in charge of dynamic driving tasks of a single-vehicle, and (*ii*) a central information system (IS) (further in the case referred to as '*central system*') that provides infotainment service to passengers and help them to set interaction with the vehicle. The systems can be either managed by the same or different service providers.

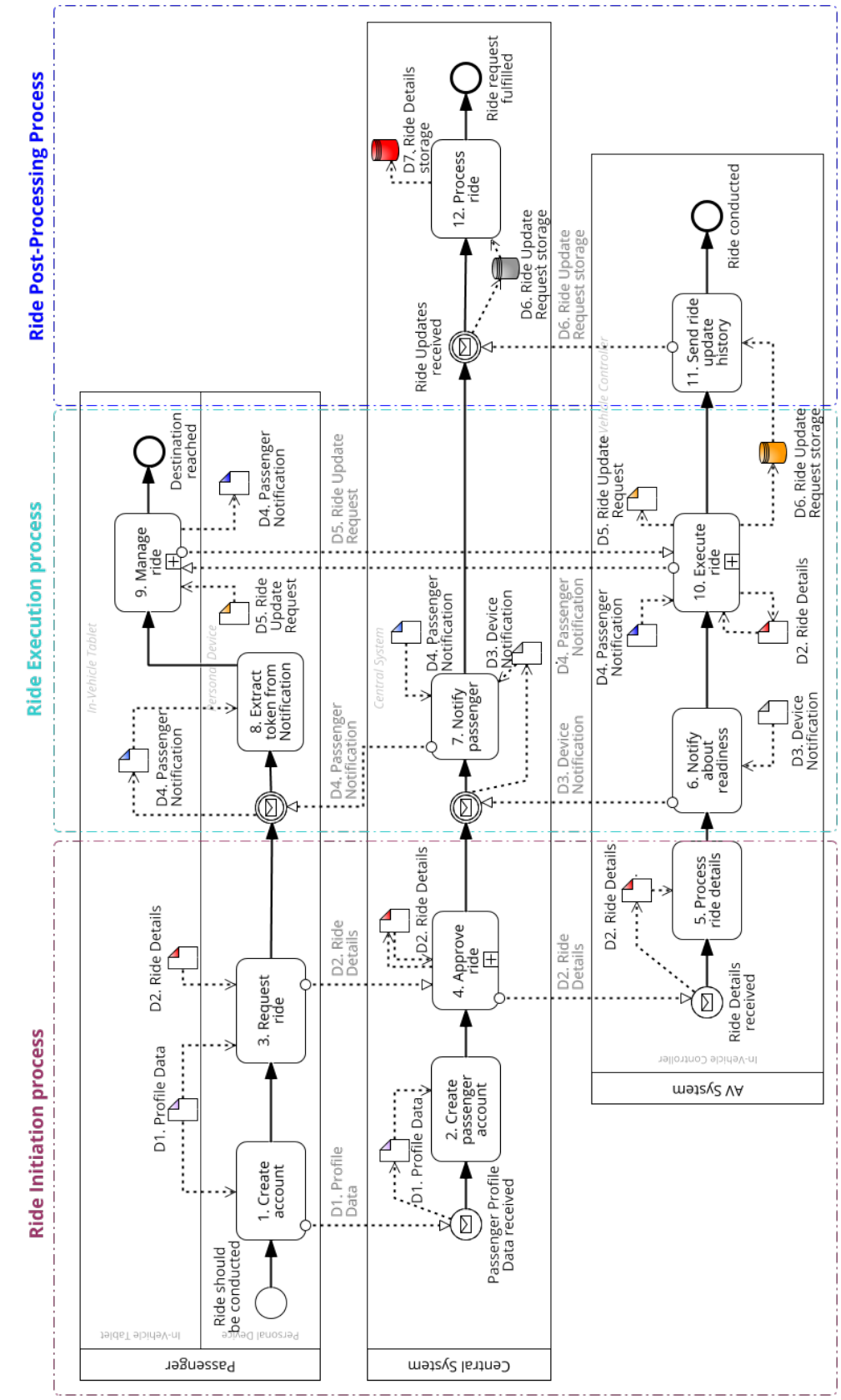

Figure 5. Ride Fulfilment business process model

Case Analysis Procedure The thesis demonstrates the results of the case study. The case study consists of two steps procedure (see Fig. 6) with respect to the stated research questions.

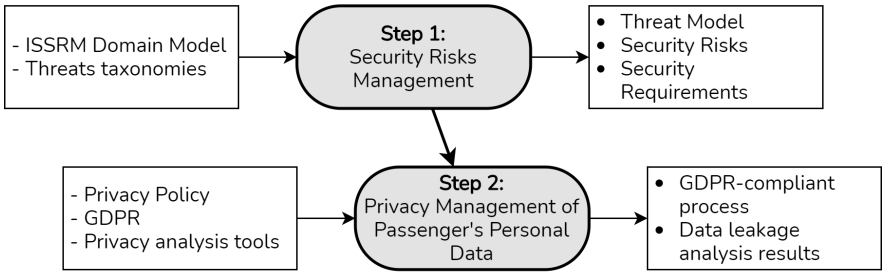

Figure 6. Case Analysis procedure

Step 1 of the research aims to answer RQ1, analysing how to secure information during the Passenger–AV interaction. Therefore, we focus primarily on the part of the Ride Fulfilment process where the Passenger is actively involved in the interaction with a vehicle - Ride Execution sub-process. Considering security risk management goals, this step of the analysis results in the approach for ensuring information security of organisational assets. Thus, we show how the CIA triad can be ensured for the case scenario.

Step 2 of the research aims to answer RQ2, analysing how Passenger's personal data privacy can be guaranteed. While 'privacy' is a broader concept than 'information security' because the former imposes requirements above and beyond the latter one, the focus of Step 2 differs. Firstly, only assets that contain sensitive Passenger's personal data are considered: (*i*) *Passenger Notification*, which contains the current Passenger's geolocation; (*ii*) *Ride Details*, which stores Passenger Notification and info about the Passenger, the selected to pursue route and, consequently, future geolocation; (*iii*) *Ride Update Request*, which reflects changes of geolocation comparing to the original route. Secondly, as the local legislation regulates privacy, the GDPR requirements for personal data processing are considered as the primary source of requirements. It should be noted that it is not enough to investigate the Ride Execution process where such sensitive personal data as the location is manipulated to ensure data privacy. To analyse how personal data is handled, we need to enlarge the scope comparing with Step 1 and understand how the Passenger-related data is initially captured and processed afterwards. Therefore, to address RQ2, we examine the whole Ride Fulfilment process. We conduct the tool-supported analysis to improve the accuracy of privacy analysis results and illustrate how the organisations can benefit from the chosen tools.

2.6 Summary

In this chapter, the background of the research is presented. We have defined two main areas that address passenger's data protection: security risk management and privacy protection. The goals and methods of ensuring the security criteria of organizational information and personal data are described. An overview of autonomous vehicle systems and their components is presented. Finally, the chapter describes a case study design used in the thesis for analysis by applying theoretical security risk management and personal data management techniques.

3 Security Risk Management

In this chapter, the information security risk management for the case scenarios is described. This chapter aims to provide readers with the necessary background about threat modelling, including common threats taxonomies and libraries and demonstrates threat model application for the presented in Section 2.5 scenario that resulted in the derived security risks. Finally, the possible security measures to address risks are presented in the form of security requirements. Thus, this section answers the question **RQ1**.

3.1 Research Method

This chapter aims to investigate *how can information security risks in Passenger-AV be managed*. We are using the ISSRM domain model [41] as a baseline for guiding the security risks and requirements definitions to address the RQ1. Therefore the sub-questions correspond to the domain model groups:

SRQ1.1: What assets should be protected in the Passenger–AV interaction? By answering this question we identify the assets present in the researched interaction and the security criteria that should be assured.

SRQ1.2: What are the security threats in the Passenger–AV interaction? After assets identification in SRQ1.1, we define their vulnerabilities, the security threats that target them, and the risks that should be managed.

SRQ1.3: What are the security requirements to mitigate security threats in the Passenger–AV interaction? Answering this question, we define how the defined in SRQ1.2 security issues can be treated by eliciting security requirements.

The research process is depicted in Figure 7. Two parallel processes are conducted: theoretical artefacts development based on the literature review and the case analysis that illustrates the derived artefacts' application. Thereby, the thesis demonstrates the results of the applied exploratory research of the under-researched case of Passenger–AV interaction.

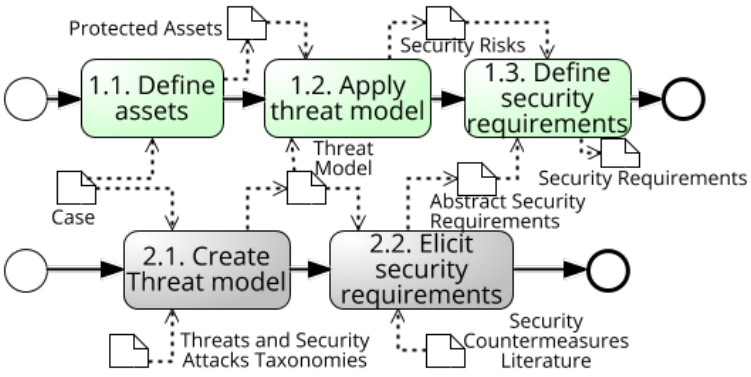

Figure 7. Security risk management: research method

The activity *2.1.* answers the question *SRQ1.1* by defining the assets that should be protected based on the designed case. The business assets are extracted from the data structure of the researched system. Ergo, UML class model (see Sec. 3.2 and Figure 8) is created to define the key data entities

and dependencies. Having the flows depicted in the business process diagram and data entities dependencies enables us to identify protected assets and security criteria.

For answering the question *SRQ1.2*, the threat-driven approach is applied for defining the risks for the scenario. The attack libraries, taxonomies and the threat modelling framework are used to determine the interaction threat model on the step *1.1* (see Sect. 3.3.1). After, the threat model is applied to the defined protected assets (activity *1.2*), resulting in the developed risks (see Sec. 3.3.3).

For mitigating the defined set of threats and risks, security countermeasures are defined. Thereby, Section 3.4 answers the question *SRQ1.3* demonstrating the elicited security requirements (activity *2.2*) based on the security countermeasures literature. Moreover, the application of the abstract requirements to the security risks (activity *2.3. Define security requirements*) results in the concrete security requirements for the scenario.

3.2 Protected Assets

This section seeks to answer the **SRQ1.1** by defining the assets that should be protected. In the Passenger–AV interaction, *business assets* (BA) are presented by transmitted data, vital for the proper processes flows. The *system assets* support these business assets and are responsible for generating, manipulating, and storing new BAs. The *security criteria* of a business asset are defined by security objectives, which describe the security needs of a system. CIA (Confidentiality, Integrity, Availability) triad forms the main security criteria which can characterise business asset.

The general data structure of the system is depicted in Figure 8 in the form of a UML class diagram.

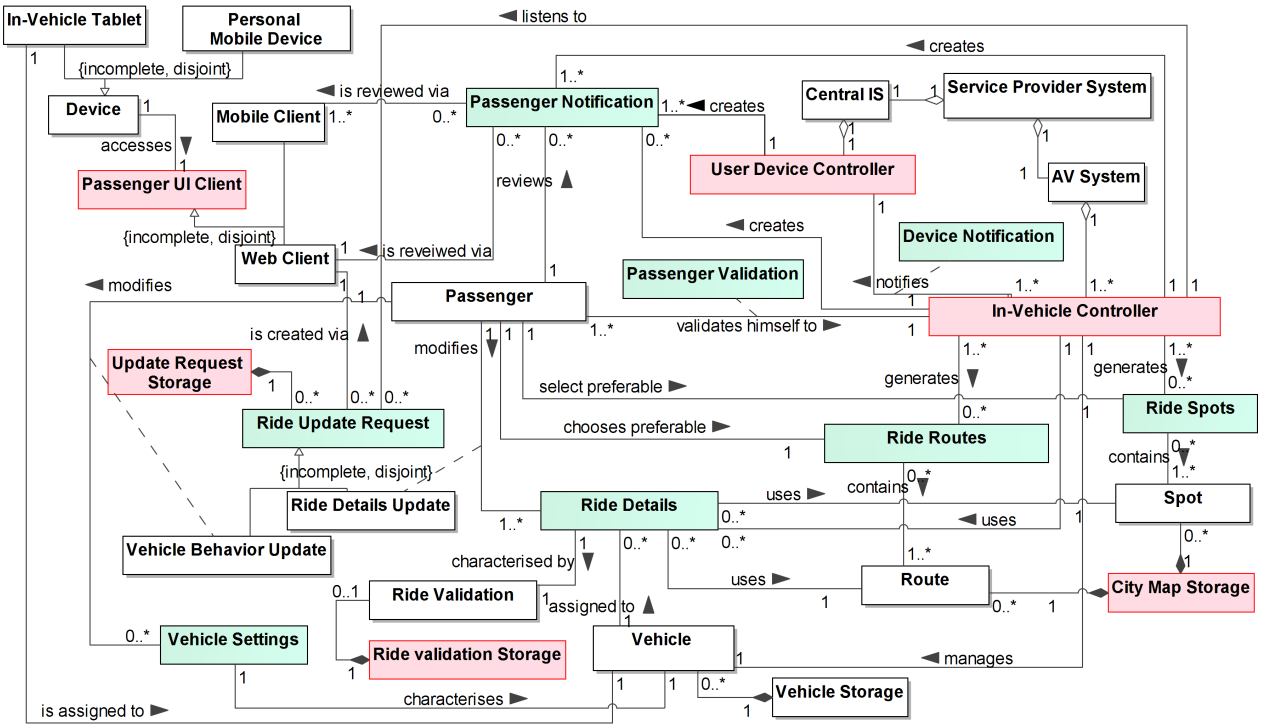

Figure 8. Data structure of the system (green entities - business assets; red entities - system assets)

As we can see from Figure 5 and Figure 8, *Ride Details* is a central asset used by *In-Vehicle Controller* for the ride execution. This entity contains such fields as a starting point, and a destination of the ride, collects information about *selected routes*, *ride spots* to get off, involved in the ride *vehicle*. Also, it contains the reference to the entity, which corresponds to a Passenger that stores the personal data. Meanwhile, *Passenger Validation* asset contains credentials that a Passenger should use to start a dialogue with the system and, consequently, start the ride. As a result, the *Ride Details* asset is a central entity for providing infotainment service to the Passenger. Figure 9 presents the defined security criteria in terms of CIA triad for the identified business assets:

- *confidentiality*: the criterion defines access control so that the business assets can be accessed only by an authorized person;

- *integrity*: the goal of the criterion is to protect a business asset from unauthorized tampering to ensure the reliability of assets and accurate decisions within the system;

- *availability*: the criterion ensures the ability to access the business asset on demand so that the system can execute normal functioning.

Protected Assets	Confidentiality	Integrity	Availability
Ride Details	●	●	●
Device Notification	○	●	●
Passenger Notification	●	●	●
Passenger Validation	○	●	●
Ride Routes	○	●	●
Ride Spots	○	●	●
Ride Update Request	○	●	●
Vehicle Settings	○	●	●

Figure 9. Security criteria of the protected assets

Now let us consider how the system assets are organised. A service provider system consists of two main components: Central IS (also called as '*Central System*') and AV System. *In-Vehicle Controller* represents the back-end part of the AV system, and it is in charge of accessing data storage, conduction most of the calculation, and data manipulation functions. *User-Device Controller* is a back-end part of Central IS. *Passenger UI Client* in the observed system represents the front-end part and is separated into *Web Client* and *Mobile Client*, which correspond to the front-end parts of AV System and Central IS, respectively. Thus AV system communicates with a Passenger via Web Client opened on the In-Vehicle Tablet, while Central IS interacts with a Passenger via a mobile app installed on the Personal Mobile Device. Other components of the architecture are application programming interfaces (APIs), which facilitate communication between the system components - *Central IS API* and *AV System API*.

The defined business assets are: (*i*) ***Ride Details*** which should be *integral, available, confidential* by means of *In-Vehicle Controller, Web Client*, and *Ride Validation Storage*; (*ii*) ***Device Notification*** which should be *integral* and *available* by means of *In-Vehicle Controller, User-Device Controller*, and *Transmission channel* (used for AV System to Central IS interaction) along with *Central IS API* and *AV System API* which ensure *integrity* and *availability* of the asset; (*iii*) ***Passenger Notification*** which should be *integral, available, confidential* by means of *User-Device Controller, Passenger UI Client, In-Vehicle Controller*, its *API* and *Transmission channel*; (*iv*) ***Passenger Validation***, *integrity* and *availability* of which are ensured by *Passenger UI Client, In-Vehicle Controller, AV System API*;

(*v-vi*) ***Ride Routes*** and ***Ride Spots***, *integrity* and *availability* of which are ensured by *Web Client, In-Vehicle Controller, City Map Storage* and *Transmission channel* used for In-Vehicle Controller to Web Client interaction; (*vii*) ***Ride Update Request***, *integrity* and *availability* of which are ensured by *Web Client* together with *AV System API*, *In-Vehicle Controller*, and *Transmission channel* used for In-Vehicle Controller to Web Client interaction; and finally (*viii*) ***Vehicle Settings***, *integrity* and *availability* of which are ensured by *In-Vehicle Controller*.

3.3 Security Risks Identification

The current subsection answers **SRQ1.2** by reviewing the threat modelling approaches and taxonomies to identify a threat model for the Passenger–AV interaction scenario. After, we show how security risks are derived from the composed threat model.

3.3.1 Threat Modelling

As we have mentioned before, information security risks are primarily defined by the attacks an adversary employs to target system assets. Therefore, the threat-driven approach for risks identification is defined as the common in [35, 12] for guiding security risk management. The threat model should be defined as a primary step for risks identification. This section reviews the common threat taxonomies and attack libraries focused on the attacker's tactics, techniques, and procedures (TTP) [54]. Surveyed threats repositories are enterprise-neutral and technically focused as they do not put any limitations on a specific enterprise, its architecture, or assets but instead concerned with the overall technological environment.

STRIDE The STRIDE approach of threat modelling [55] aims to help software developers identify which type of attacks the product can be harmed by. STRIDE stands for the five threat groups: Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, and Elevation of privilege. Initially, the approach was designed for eliciting system security threats and generic threat vectors to be considered, but not for categorizing threats by itself. Therefore it is beneficial to apply STRIDE at the beginning of conducting ISSRM during to brainstorm potential risks [12].

CAPEC The Common Attack Pattern Enumeration and Classification [56] (CAPEC) is a comprehensive, community-created catalogue of attack patterns. It defines the informal taxonomy of attack-pattern classes and provides the formal description of each attack class. The taxonomy is organized hierarchically based on its domain and mechanisms of attack, specifying the vulnerabilities it addresses. Furthermore, CAPEC is supported by references to the targeted vulnerabilities and possible mitigations.

ATT&CK Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK) framework [57] is a globally accessible knowledge base of adversarial techniques which helps to classify attacker's actions for each particular platform (e.g., Windows, Android). The MITRE Corporation created the framework as a baseline for threat modelling in different domains and sectors. ATT&CK is focused on techniques in the context of tactics an adversary wants to apply to attack a specific component or endpoint. The presented techniques an adversary may use are supported with the procedure

examples, system requirements for implementing the tactic (e.g., platform, required permissions), possible detection methods, and mitigations. Moreover, most of the techniques are mapped to the corresponding attack pattern in either CAPEC or Mobile Threat Catalogue. The attacks in ATT&CK are also supported by incidents examples and a set of measures for preventing the attack execution.

OWASP Top Ten The list of vulnerabilities provided by the Open Web Application Security Project (OWASP) [58] is considered a starting point for developing secure web application software focused on defensive mechanisms and controls. OWASP group creates top lists of vulnerabilities every other 3-4 years. The list defines the most severe security risks for web applications. The latest Top Ten vulnerabilities version was created in 2017, and this version is used as a baseline in this thesis. However, the approach does not consider the prospect of a threat agent or any application implementation details. For that reason, the current list should not be used as a strict guide for handling ISSRM for any particular web application. Thus, for each specific case, a threat agent, assets, and corresponding impact should be considered besides, respectively.

In the further subsection, we demonstrate how the reviewed threats taxonomies and libraries can be used for deriving a threat model on the example of the Passenger–AV interaction scenario. A similar approach of threat modelling is also considered in [14], where authors used STRIDE to support ISSRM in security risk management in e-commerce systems.

3.3.2 The threat model for Passenger–AV interaction

To define a threat model, we consider the case scenario presented in Section 2.5. Firstly, we have used the STRIDE approach to brainstorming and elicit potential attack vectors in the context of the AV system architecture and the Ride Fulfilment process. Then we have reviewed the OWASP Top Ten list to define the possible vulnerabilities of the web application component of the AV system. Afterwards, we have inspected CAPEC and ATT&CK taxonomies for identifying concrete attacks by which the AV system can be harmed. When selecting attacks from CAPEC and ATT&CK, we have paid attention to the prerequisites of an attack implementation (e.g., required adversary's skills), targeted vulnerabilities, tactic and examples of attack execution. Based on the selected attacks and vulnerabilities relevant to the Passenger–AV interaction, we have formed a threats model. The described process has been iteratively executed for In-Vehicle Controller and the Central IS.

Figure 10 illustrates the derived threat model for the Passenger–AV interaction. The model contains 17 threats that an adversary can exploit during the Ride Execution process. The threats are organised into six groups of threats from STRIDE approach. Each threat is supported by the reference to the source it is elaborated from. A detailed description of the threats (targeted vulnerability, threats agent, attack method, and potential impact) can be found in Appendix III.

Spoofing refers to identity spoofing attacks where an attacker pretends to be a legitimate passenger. An attacker uses the obtained credentials to violate the system's *authentication* mechanism (ST1). In the case of ***tampering***, an attacker intentionally modifies a system, network, its behaviour, or the data to violate their *integrity*. These threats target data storage (TT1) and software source code files (TT3) used during the ride execution and are critical to general trip safety and data reliability. As the Passenger–AV interaction includes communication between few separate entities (AV system and Central IS), API parameters can be manipulated for changing the normal entities communication (TT2). ***Repudiation*** attacks target the business layer during which the system cannot track and log actions accurately. As a result, the system claims that the activities were not done even if they were,

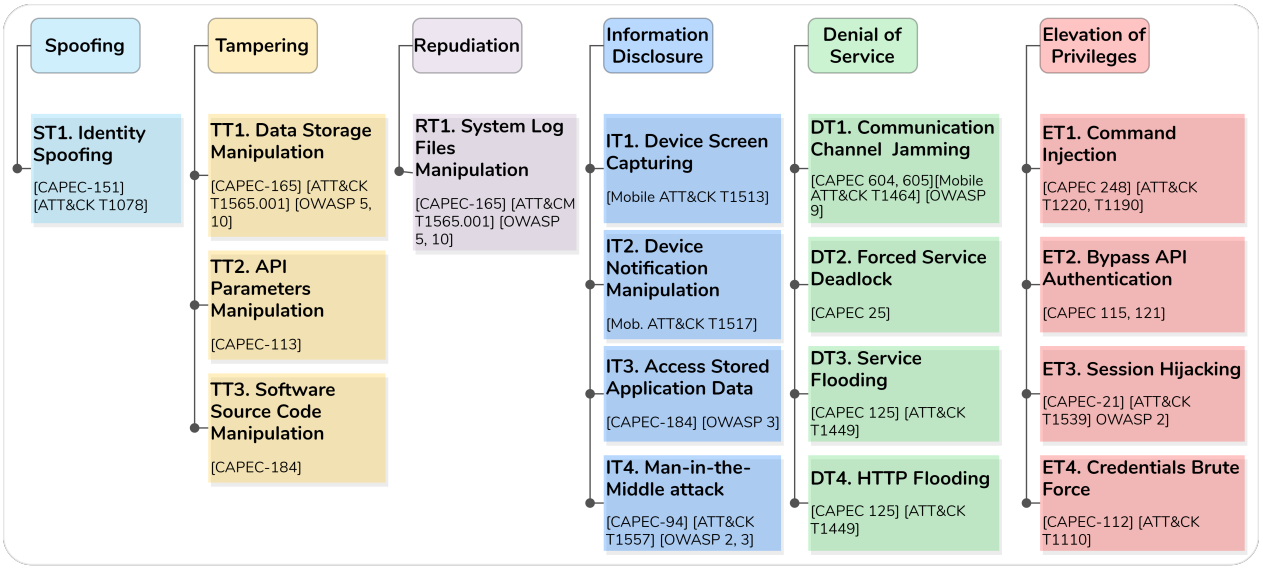

Figure 10. The threat model for the Passenger–AV interaction

or vice versa. By manipulating the system log files that keep track of both passenger's activities during the ride and the data about the driving task execution, an attacker can influence the current and the future ride that uses the historical data (RT1). ***Information Disclosure*** groups the threats in which the confidentiality of the data is violated by providing access to it to someone who is not supposed to have access. It refers to accessing data while it is stored locally (IT3), displayed on the mobile device to a passenger (IT1, IT2), or in the transmission between systems or their components (IT4). Such attacks intend to gather information required for further attacks. ***Denial of Service*** attacks are focused on consuming resources needed to provide service to a Passenger, and as a consequence, the *availability* of the information is violated. The threats target either the communication channels' resources (DT1 and DT4) or computational resources (DT2 and DT3). ***Elevation of Privileges*** threats refer to allowing an attacker to have authorisation permissions that he was not supposed to have, thereby violating the system's *authorisation*. It can be achieved either using the obtained legitimate credentials (ET3 and ET4) or by a more sophisticated manual bypassing the existing authentication mechanisms (ET1 and ET2).

3.3.3 Security Risks

Having a threat model defined, security risks for the observed scenario can be derived. The process of deriving risks from the threat model is iterative, and the described later process should be repeated for each threat in the model.

Firstly, one needs to look at a vulnerability the threat is based on. Then, system assets should be examined to find those which have such a vulnerability. For the system assets that has such a vulnerability, one should instantiate the threat which targets this vulnerability. Then one needs to define how this threat can exploit the system asset's vulnerability by answering the following questions: (*i*) which business assets the analysed system asset supports; (*ii*) which security criteria are negated; (*iii*) what impact the threat implementation have. In this way, a risk is defined and ready

for its assessment and further decision about its treatment.

As a result of repeating the described procedure, we have identified a set of security risks (so-called, *risks model*). The risk model includes 22 information security risks that can take place in the Service Provider System and impact the business assets. The complete model can be found in Appendix IV. Figure 11 aggregates the defined impact of the risks on protected assets.

Business Asset	Confidentiality	Integrity	Availability	№ influencing risks
Ride Details	SR1, TR6	TR6	TR6, RR1	3
Device Notification	-	TR4, TR5,	DR2	3
Passenger Notification	SR1, TR5, IR1, IR2, IR3, IR4, IR5, ER4	TR4, TR5, TR6	TR6, IR3, DR1	11
Passenger Validation	-	TR2, ER1	DR1, DR5	4
Ride Routes	-	TR1, TR6	TR6, DR1, DR3	4
Ride Spots	-	TR1, TR6	TR6, DR1, DR3	4
Ride Update Request	-	SR1, TR3, TR6, ER2, ER3	TR6, DR1, DR4	7
Vehicle Settings	-	TR6	TR6	1
№ influencing risks	9	10	8	

Figure 11. The resulted impact of derived security risks on the protected assets

Among the derived risks, 13 risks are targeting Passenger Notification, and 8 out of 22 risks are targeting confidentiality of Passenger Notifications. Furthermore, some risks include harm to the system components, which may result in getting access to any sensitive data visible to the system. Table 1 contains an example of the derived security risks – namely, *IR5: Man-in-the-Middle (MitM) attack* which targets *Passenger Notification*. According to CAPEC, the MitM attack requires medium skills level of an attacker but has a high impact, as it enables an adversary to conduct further attacks on the system.

To illustrate the attack implementation, we are using the security extension to BPMN [59], which supports the ISSRM domain model. In Figure 12, we see an attacker as an additional entity (pool) that intercepts in the transmission channel aiming to define when the AV with the Passenger reaches the desired place on their route.

The Man-in-the-Middle attack (threat IT4) primarily aims to negate the confidentiality of Passenger Notification. However, effective delivery enables an attacker to conduct a set of further attacks that already may target the vehicle's functions (e.g., dynamic driving tasks). Therefore, during the risk assessment and decision about its treatment, it is vital to consider the direct impact and the impact of the risks that may be provoked by it — consequently, a threat-driven approach helps make requirements prioritisation (e.g., like it is proposed in [60]).

Table 1. Security risk IR5: Man-in-the-Middle attack description

Business Asset	Passenger Notification
System Asset	Transmission channel used for Web-Client to In-Vehicle Controller communication
Vulnerability	The **transmission channel** used for Web-Client to In-Vehicle Controller communication is not protected with mutual authentication; The communication between system components is conducted without data encryption.
Threat	IT4. An attacker places himself in the communication channel between Web-Client and In-Vehicle Controller to passively or actively listen to the transferred data flows.
Impact	Compromises **confidentiality** of **Passenger Notification**
Risk	An attacker places himself in the transmission channel between Web-Client to In-Vehicle Controller to passively listen to the transferred data flows and exploit the lack of data encryption that leads to compromising the confidentiality of Passenger Notification.

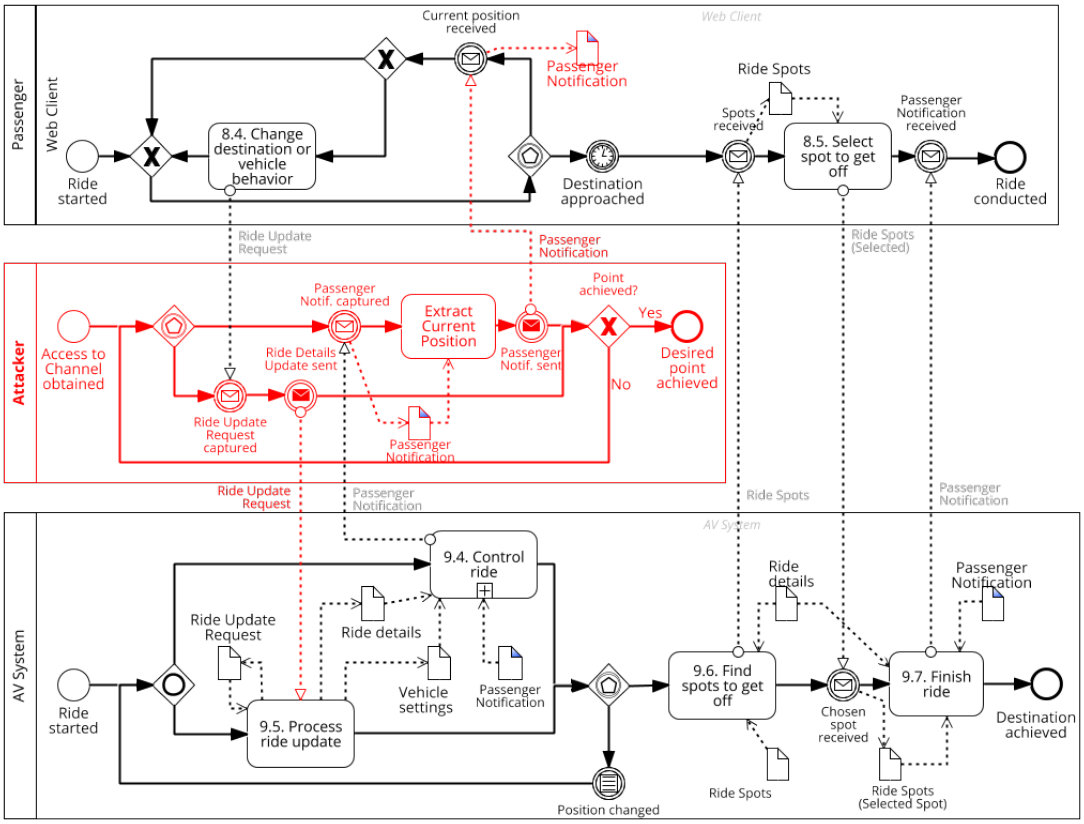

Figure 12. Security risk IR5: Man-in-the-Middle attack execution

Finally, we want to highlight that the list of security risks is not final in the context of described system architecture, and some risks may have been omitted due to the lack of the author's expertise in the system vulnerabilities and the low level of details of system components. Furthermore, when

implementing the researched scenario in real life, an organisation may use system assets with different vulnerabilities comparing to the mentioned in the thesis. Therefore, the derived risks model should be refined for each implementation of Passenger–AV interaction considering the business process and system characteristics.

3.4 Risks Mitigation

The current subsection describes the process of security requirements elicitation based on the defined threat and risk models. Here we presents a list of the derived security requirements that answers the last sub-question (**SRQ1.3**).

3.4.1 Security requirements elicitation

According to [61] and [34], a *security requirement* is a condition of the domain environment that should be met in order to mitigate one or more security risks and utilising security controls implemented in the system. In [62], NHTSA presents the recommended guideline to the automotive industry for the vehicle's electronic architecture. It is intended to improve vehicle cybersecurity by implementing security controls. They also emphasise the necessity of using information technology security suite and standards (ISO 2700x series, CIS [63]). Similarly, the report by ENISA [6] contains a set of good practices for smart cars. It stresses that for conducting information security management, it is crucial to use the standards mentioned above along with SAE J3061 [64] and NIST 800-53 [40]. Meanwhile, IPA proposes the guide [10] for achieving a security level in the automotive systems. They highlight security management by implementing the security function design (in the sense of encryption, authentication, and access control), which should be enhanced with secure coding, security testing, and user training.

The threat-driven requirements elicitation approach supports security requirements categorisation on three groups: (i) *preventive*; (ii) *detective*; (iii) *corrective*. The analogue taxonomy of security countermeasures for the AV defence is presented in [48]. Besides defining the architectural design, the requirements classification could be used in the requirements prioritisation. Thus, the type of requirements may be considered during the impact assessment; for example, preventive or detective controls may have a higher impact score than others based on the organisational risk treatment strategy.

3.4.2 Defined abstract security requirements

For mitigating the defined threats in the Passenger–AV scenario, we consider the security controls from the standards mentioned above and libraries. The requirements are later defined using the inductive approach from the found controls. Using [58], [56], [63], [40], [57] as the main primary sources, we have elicited 56 requirements which are supported with their possible implementations (i.e. security control components), organised in groups of the correspondent treated threats. The full list of elicited requirements can be found in Appendix V, while Table 2 represents few examples.

Table 2. Security countermeasures identification (P - preventive, D - detective, C - corrective); here the term "*system*" refers to the system(s) used in AV for the Ride Fulfilment

Security Requirements	Class	Security Control Components
IT2. Device Notification Manipulation.		
IT2.R1. The system should protect sensitive data provided to the mobile app.	P	**IT2.C1.** Include only non-sensitive data in the app notification text [57].
IT2.R2. The system shall guide users to set particular configuration settings on the mobile devices used for interaction with the system.	P	**IT2.C2.** Advise not to grant consent for device manipulation [57].
IT4. Man-in-the-Middle Attack.		
IT4.R1. The system should verify integrity of the transmitted data.	D	**IT4.C1.** Cryptographic hash functions: message authentication code (MAC) algorithms [48], [65], digital signature, and checksums [65].
IT4.R3. The system should ensure the confidentiality of transmitted information.	P	**IT4.C4.** Cryptographic mechanisms: SSL/TLS protocol [63], IPSec protocol suite [65].
		IT4.C5. Advanced encryption standard (AES) to encrypt wireless data in transit [63].
IT4.R5. The system should authenticate device before establishing connection.	P	**IT4.C9.** Bidirectional cryptographically based authentication [65].
IT4.R6. The system should follow the wireless capabilities policies.	P	**IT4.C10.** Usage of wireless networking capabilities only for essential functions [63], [65].

As can be seen, along with the elicited system requirements, organisational policies and user guidelines clauses are derived. It supports the claim about the complexity of the researched scenario. Thus, the key to managing it security risks in the Passenger–AV interaction lies in the interception of system security and human behaviour management.

3.4.3 Security Requirements

The current subsection contains a list of derived security requirements for the AV ride-hailing service provider system. We have instantiated the corresponding abstract requirements derived earlier from the literature for eliciting concrete security requirements for the researched case. Hence, this subsection demonstrates discovered security requirements for the Bolt system.

Full set of security requirements can be found in Appendix VI. Figure 13 aggregates the derived requirements for Bolts system. We can be see that 18 of the requirements aim to deliver the confidentiality of the protected assets, while 10 authentication and 11 authorization requirements also indirectly ensure limited access to the system assets. Only 6 requirements address the availability of assets as the corresponding threats are hard to prevent. In total, we have formulated 64 security requirements for the Bolt system, which aims to mitigate the derived risks.

Addressed Security Property	Mitigated risks	Number of requirements
Authentication	1	10
Ingerity	6	13
Non-repudiation	1	6
Confidentiality	5	18
Availability	5	6
Authorization	4	11
	Σ	**64**

Figure 13. The derived security requirements analysis

To illustrate how the elicited requirements can be addressed in the presented system, Table 3 contains examples of the security requirements, which mitigate risk IR5 in the Bolt system.

Table 3. Security countermeasures for IR5 mitigation

Security Requirement	Security Control Components
R1. *In-Vehicle Controller* should identify unauthorized connections to the local area network.	**IT4.C3.** Authenticated application layer proxy for the network traffic that comes goes to or from the Internet [63].
R2. *In-Vehicle Controller* should ensure the confidentiality of transmitted via *Transmission Channel* information.	**IT4.C4.** Cryptographic mechanisms: SSL/TLS protocol [63], IPSec protocol suite [65].
	IT4.C5. Advanced encryption standard (AES) to encrypt wireless data in transit [63].
R3. The service provider who owns the vehicles should control physical access to *transmission channel* within organizational facilities.	**IT4.C6.** Wiretapping sensor [65].
	IT4.C7. Locked wiring closet [65].
	IT4.C8. Protection of cabling by conduit or cable trays [65].
R4. *In-Vehicle Controller* should authenticate mobile device before establishing connection.	**IT4.C9.** Bidirectional cryptographically based authentication [65].
R5. *Service Provider System* should follow the wireless capabilities policies.	**IT4.C10.** Usage of wireless networking capabilities only for essential functions [63],[65].

The provided controls should not be considered as must-to-implement, but instead, they give system developers and system owners an understanding of possible ways to fulfil the requirements. It should be noted that the requirements in Table 3 are not prioritised yet, and for the usage in the system development lifecycle, the prioritisation must be done based on the prior risk assessment, existing system architecture, and the costs of implementation.

3.5 Validation Design

As part of the research, we employ a qualitative analysis method to validate the proposed security requirements elicitation approach. We have presented preliminary results on the seminars with the subject-matter experts. The derived threat model and security risks have been presented at the seminar for PhD students and researchers of information security in the Institute of Computer

Science, University of Tartu. There have been 6 participants in the seminar. The participant have expressed support of the presented results.

Additionally, further quantitative analysis is needed to validate the proposed approach. Such a quantitative analysis should be done by making the requirements comparison. The SQUARE [66] and SREBP [67] methods for eliciting security requirements can be used as an alternative as these methods are also applicable at the early stages of system analysis and design [68] but differ on the approach (risk-driven and asset-based, respectively).

After defining a set of security requirements for the Ride Fulfilment business process using an alternative elicitation method, one needs to conduct a qualitative and quantitative assessment of both requirements sets. Each received set should be analysed on the security requirements categories to identify the requirements categories' coverage. The coverage should be assessed by analysing how many asset's attributes can be delivered by implementing the elicited requirements. Such requirements qualitative analysis enables the quantitative assessment of requirements categories and further comparison of the results.

To avoid the carry-over effect, we recommend conducting requirement elicitation using an alternative method by a person not familiar with the current thesis's results. It should be noted that the number of elicited requirements should not be used for the assessment of the methods.

3.6 Summary

In this chapter, security risk management is discussed. The threat modelling approach for deriving security risks and security requirements is presented. Bolt's information system architecture is identified to cover security threats, and the business assets to be protected are recognised. Next, we have selected four state-of-the-art threats taxonomies and libraries: STRIDE, CAPEC, ATT&TK, and OWASP Top Ten. Based on the taxonomies and libraries, the threat model is created. The risks for the defined assets in the context of the case are identified. The possible mitigation measures are formulated in the form of security requirements. The chapter also presents preliminary validation results, and the design further required validation of the requirements elicitation approach. In the next chapter, the personal data management in Bolt systems that participate in the Ride Fulfilment process is discussed. While in the current chapter, we have discussed how to manage uncertainties that may affect the overall security of the organisation's owned information, the next chapter discusses how the collected by the organisation personal data should be managed.

4 Privacy Management of Passenger's Personal Data

In the previous chapter, we looked at how information security in the selected scenario can be ensured. Thus, we defined how to fulfil one of the principles needed to protect data privacy. This chapter examine how the other data protection principles can be met to ensure passenger's data privacy, assuming that the information security is already assured. Hence, this chapter shows how meeting local legislation requirements can be checked. The usage of privacy-enhancing technologies for assuring personal data protection in the Passenger-AV interaction is illustrated. Finally, the chapter demonstrates the usage of DPO and Pleak tools to explain the benefits of tool-supported analysis.

The results presented in the chapter answer the second research question (**RQ2**): How can the privacy of Passenger's personal data be managed in the Passenger-AV interaction?

4.1 Research Method

The following sub-research questions should be answered to address RQ2:

SRQ2.1: What are the measures to protect personal data according to GDPR? By answering this question, we employ tool-supported GDPR compliance check and create two business process designs using different technical measures to comply with the GDPR.

SRQ2.2: How to compare privacy-enhancing technologies in the context of the business process? After defining the protection schemes that implement requirements defined in SRQ2.1, we analyse the personal data visibility to define how PETs can be used to protect the business process from data leakages.

To answer the stated research questions, we use presented in Figure 14 research method.

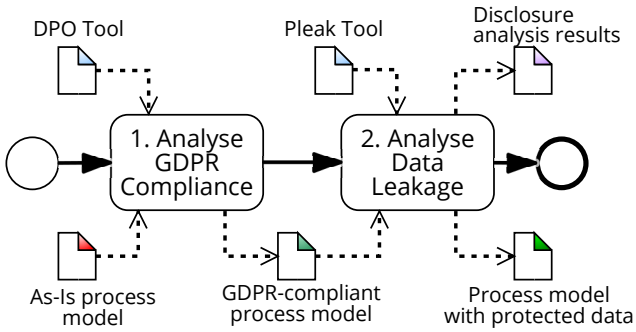

Figure 14. Privacy of Passenger's personal data: research method

To address SRQ2.1, firstly, we analyse the compliance of the existing (i.e., As-Is) process with the GDPR requirements. For this reason, we defined the As-Is process model from the scenario (see Sec. 2.5) and then we extend it considering Bolt's privacy policy. DPO tool is used for conducting the analysis. The tool should be applied for defining the non-compliance issues. Based on the provided results, the redesigned GDPR-compliant process model can be produced. It should be noted that GDPR does not strictly rule implementation of the security measures and provides high-level requirements on the data protection principles. Therefore, there can exist more than one GDPR-compliant business process for one non-compliant As-Is process. The main difference

between the compliant process implementations is in the selected PETs used for the personal data protection. To demonstrate this, we define two process designs that implement different PETs.

In the next step, we address SRQ2.2 by analyzing data leakages. For this purpose, we analyse the GDPR-compliant process using the Pleak tool. This step of the research enables a comparison of the PETs for the scenario. The tool identifies possible data leakages providing the results of disclosure analysis for the created process models. While the first step of personal data management - GDPR-compliance check - seeks to answer how personal data should be protected according to the local legislation, the data leakages analysis complements the previous step's results by analysing the selected data protection technologies. Taken into account the disclosure analysis results, we propose a process model that delivers better data protection and compliance with the local legislation. The disclose analysis results can be used late by the business process owner for the information security risk management, as the results give clues on the existing vulnerabilities within the implemented process and system.

4.2 GDPR Compliance Analysis

In the current section, we analyse the Passenger–AV interaction scenario by checking its compliance with the EU data protection regulation - GDPR. For a personal data controller to comply with the GDPR requirements, the gaps in the systems implementation and organisational policies should be identified and filled. We conduct tool-supported analyses by employing an automated GDPR compliance check tool. Particularly, we demonstrate usage of the DPO tool to check the business process compliance to the legislation.

As described in Section 2.1.3, the first step of the tool requires as an input the As-Is business process model and the specified key terms from the GDPR model. Since GDPR regulates the entire lifecycle of the data usage by a controller, we analyse the whole Ride-hailing Fulfilment process described in Section 2.5 and visualised in Figure 15. The differences between Figure 5 (used in the case design) and Figure 15 (used in this section) are the following: (*i*) the latter considers only data objects that contains personal identifiable information; (*ii*) communication between Passenger and AV system during conducting activities *9. Manage ride* and *10. Execute ride* is omitted on the process model as we analyse from the perspective of Central System, but it is present on the level of the activities implementation and should be analysed separately; (*iii*) the notations for data storage are different in the models, but the described data entities are the same; (*iv*) the asset *D4. Passenger Notification* from the case design is separated into two entities - *D4. Passenger Notification FIRST* and *D4. Passenger Notification* - to explicitly differentiate the initial notification sent from Central System and all the other notification sent by AV System. In this section, we call the business process model depicted in Figure 15 as Ride Fulfilment (As-Is) business process model.

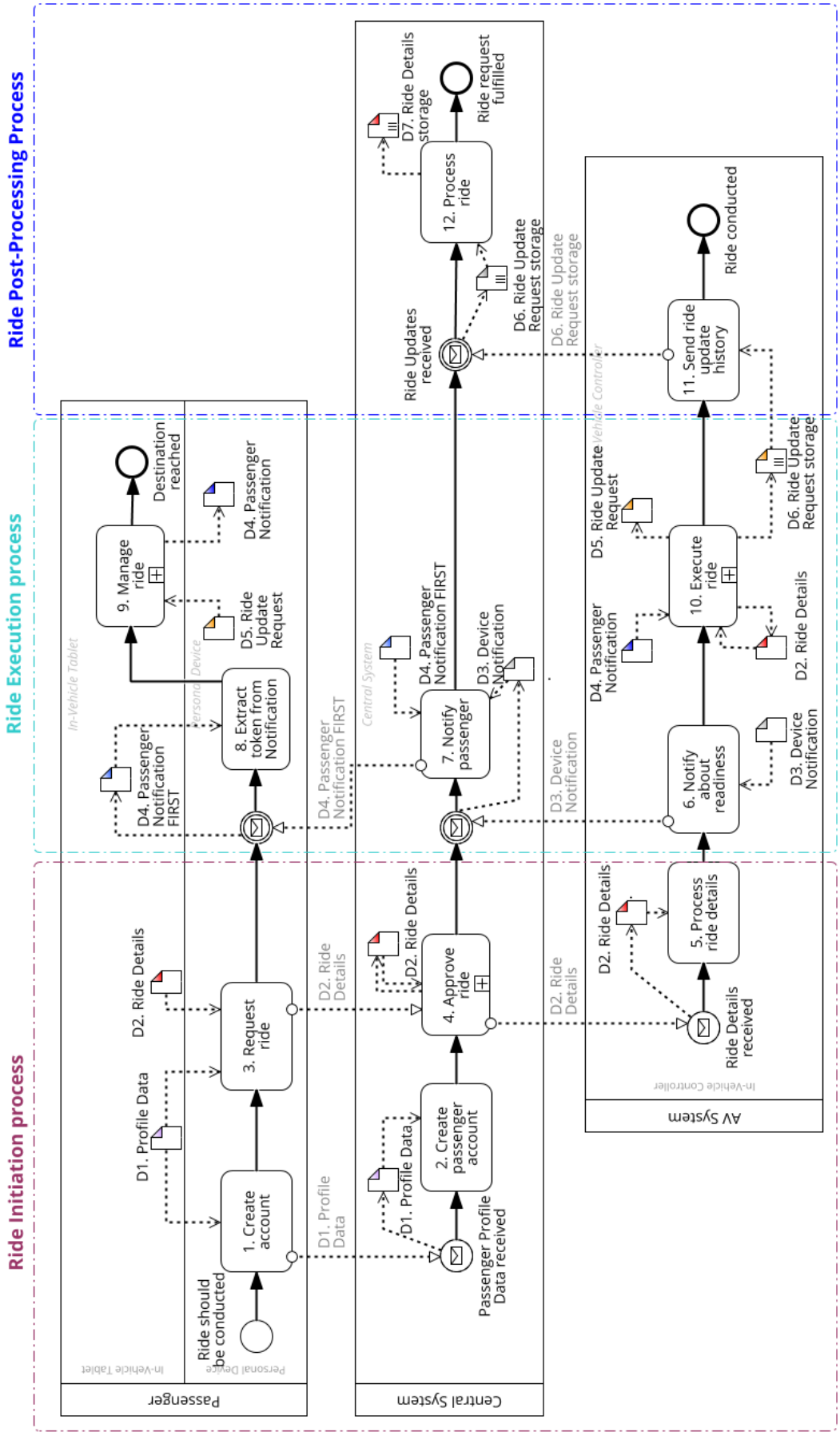

Figure 15. Ride Fulfilment business process model (As-Is)

4.2.1 As-Is Process Analysis

As the first step of As-Is Process analysis, we have supplied the initially researched business process to the DPO tool[9]. Additionally to the BPMN model, the following key elements of the process have been provided:

1. GDPR role assignment: (*i*) data subject; (*ii*) controller; (*iii*) processor; (*iv*) recipient (optional); (*v*) third party (optional);

2. *Personal data* object to analyse;

3. *Processing task* corresponding to the identified personal data;

4. Purpose of processing and consent: (*i*) *category* that the personal data; (*ii*) *legal ground* under which the personal data is collected; (*iii*) whether the data subject's consent collected prior to the processing tasks performed by the Controller; (*iv*) whether required information provided to the data subject prior to data collection (e.g., in the Privacy policy);

5. Processing system and technical measures: (*i*) security/privacy attributes of the processing system; (*ii*) data storage characteristics; (*iii*) secure technologies implemented; (*iv*) whether the controller implements organizational measures such as adherence to a security/privacy standard (e.g., ISO27701); (*v*) whether a record of processing is maintained.

The described in Section 2.5 business process does not consider most of the asked elements. Therefore, we have defined the first three elements groups based on the definitions from the GDPR and the business process context leaving the others not specified.

In the researched As-Is process *data subject* is Passenger, *Central System* is a *controller*, *AV system* is a *processor* for the tasks *4. Approve ride* and *11. Process ride* with *D2. Ride Details* as a personal data object. The ground for such roles assignment is that in the case of the Ride Fulfilment process, Passenger (data subject) initially make a request to a Central System (controller) about the ride conduction providing Ride Details with the starting point, destination and other information about Passenger (personal data). Thereby, Central System controls the provided personal data about Passenger. Central System asks AV System to conduct the Ride Details processing on its behalf based on the existing agreement to resolve the request. In this way, activities 4, 7, and 11 of the Central System can be grouped and considered as one activity "Resolve request" for conducting which Central System (controller) provided AV system (processor) with Ride Details (personal data) to be processed.

On the other hand, *AV system* itself can be considered as a *controller* when it conducts the Ride Fulfilment. Therefore, executing the task "9.Execute ride" AV system plays a role of a *controller*, when *Passenger* is a *data subject*, and *D5. Ride Update Request* and *D4.Passenger Notification* are personal data objects.

Additionally, In-Vehicle Tablet on which Passenger uses Web Client (a part of Central System) can be considered as a *controller*. Then, Passenger as a natural person is a *data subject*, In-Vehicle Tablet is a *controller* to which a natural person gives personal data (*D5. Ride Update Request*) for conducting the processing task *8. Manage ride*.

As a result, there are several processing tasks in the researched process on the different phases of the Ride Fulfilment. Considering Central System (including Web Client used on the In-Vehicle Tables) as one entity and AV System as a separate system, at the remaining of this Section we focus on the process from the perspective of Central System. Therefore, we assign Central System as a

[9]DPO Tool can be accessed at `https://dpotool.cs.ut.ee/`

controller, AV system as a processor, and Passenger as a data subject, and use this assignment to analyse how Central System can ensure GDPR compliance of the Ride Fulfilment process.

An example of the instantiated GDPR model for the processing task *4. Approve ride* is depicted in Figure 16.

As the tool runs analysis for one specific processing task, the compliance check has been repeated for every processing task conducted by the *Controller*. This results in the following found non-compliance issues:

1. Privacy Policy is missing (Art.13, 14 of [8]).
2. Consent is missing (Art.7 of [8]).
3. Processing systems (Central system and AV system) has missing attributes (Art. 32 of [8]).
4. Processing tasks are not being recorded (Art. 30 of [8]). The record of processing should include name, purpose, contact details, personal data category, data storage period, security measures, recipient (Art. 30 of [8]).
5. Central system as a data processor should be secured with security measures including standards (e.g., ISO27701) and concrete technologies (e.g., PETs).

As the next step, we have analysed the existing Privacy Policy for Passengers [69], found on the website of Bolt, to extend the initially researched business process with attributes mentioned in the policy.

Appendix VII contains the defined GDPR model elements required by the DPO tool. The Privacy Policy for Passengers [69] is used for defining GDPR model elements, but the legal person should confirm the correctness of the defined characteristics. Moreover, for those model elements which it is not possible to define the characteristic, we mark them in the table as such that are not met (N).

According to the same Privacy Policy for Passengers [69]: "personal data collected in the course of providing the services is transferred to and stored in the data centres of Zone Media Ltd. and/or Amazon Web Services, Inc., which are located in the territories of the Member States of the European Union". Therefore, we mark Data Storage characteristic as resolved in the table.

It should be noted that presented in Appendix VII GDPR model elements are not depicted in the As-Is process model but are present in the process. Therefore, the missing attributes and components should be added to the business process and the illustrating business process model to establish a GDPR-compliant Ride Fulfilment process in Bolt.

4.2.2 Process Redesign

To comply with the GDPR and implement the privacy by design principle, we have created two designs that differ by the implemented privacy-enhancing technologies. Both designs address the first four non-compliance issues in the same way.

General redesign aspects should be addressed primarily and are originate from requirements in [8]. First of all, the As-Is process model should be altered to illustrate that at the beginning of the process Central System provides the Passenger with a *Privacy Policy* artefact. The required attributes of the privacy policy should be explicitly mentioned in the process. Another change that should be done in the current As-Is process model is giving consent by Passenger. The consent should be extended by defining the required attributes.

A data controller should ensure confidentiality, integrity, resilience, availability of the controller system services. It means that these attributes should characterize the Central System in the researched case. Central System also should assure consent processing, pseudonymity, data

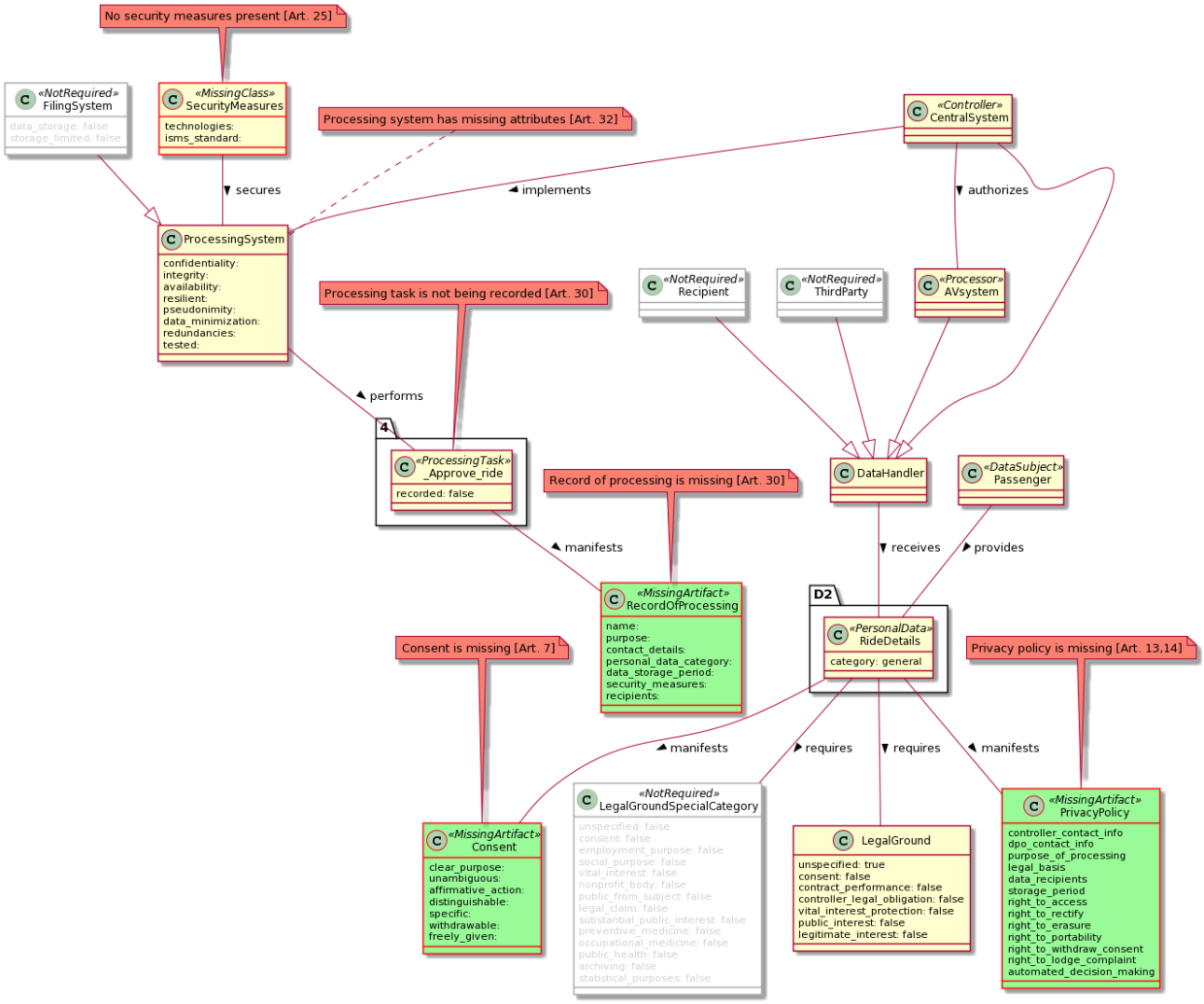

Figure 16. Result of GDPR compliance check: Ride Fulfilment process model (As-Is)

minimization, redundancies. Additionally, the system should be tested and should adopt an ISMS standard. Finally, there should be maintained records of processing. In the process model (see Fig. 17 and 19) it is described by the *Document processing of location information* activity which produce a *record of processing* artefact. The record should contain the following information: name, purpose, contact details, personal data category, data storage period, security measures, third countries transfer, recipients.

Now let us present how privacy-enhancing technologies can be implemented in the researched process. Two designs that differ in the proposed privacy-enhancing technologies - PK encryption and secret sharing - are presented below.

Design 1 implements *public key encryption* of the data object that contains personal data. Public key (PK) encryption [23] is confidentiality assuring PET that specifies data protection mechanism. The PK encryption requires a pair of keys: *public key* is used for the encryption operation to protect the data, and *private key* is used to decrypt the data. In the same paper Pullonen et al. also argue that PK computation, as an example of a fully homomorphic encryption scheme, enables privacy-aware computation. PK computation uses the encrypted data as an input and gives the encrypted data as an output. The encrypted output corresponds to the original key pair using which the input was originally protected. Hence, PK encryption, as any other encryption and data protection mechanisms, is applied to the data before its transmission from the data owner to an external party. Transmission of the protected data guarantees that it can be accessed only by the entity with a private key from the pair. Meanwhile, PK computation enables the party to conduct processing of the encrypted data as a black-box in the sense that the processing system does not have access to the original data.

The application of the PK encryption scheme is demonstrated in Figure 17, where yellow objects represent technical measure implementation and purple - other changes made in the As-Is model to resolve the non-compliance issues. In the context of the Ride Fulfilment process, *D2. Ride Details* contains sensitive data about Passenger's location and, therefore, need to be protected.

Before transmitting the data to any external system, it should be encrypted using the public key. Any intermediate processing system can conduct processing using the PK computation, and only the system that requires access to the sensitive personal data has the corresponding private key for data decryption. As a result, Passenger encrypts *D2. Ride Details* during the activity *Encrypt ride details*, then it is transmitted to *Central System* and processed using PK computation (see activity *4. Approve ride*). The resulting encrypted output is transferred to *AV system* which uses *private key A* for getting access to the Passenger's location. After reaching a destination point of the ride, *AV system* conducts post-processing of the *D6. Ride Update Request Storage* considering the changes in a route during the ride compared with the initially requested in the *D2. Ride Details*. These changes may help adjust algorithms and maps by *Central System*, but there is no need to give access to the Passenger's location history which is also stored in *D6. Ride Update Request Storage*. Therefore, during the activity *10. Send ride update history*, *AV system* encrypts *D6. Ride Update Request Storage* and transmits the protected sensitive data to *Central System*. The later one executes PK computation in the activity *12. Process ride* and stores the encrypted output data in *D7. Encrypted Ride Details Storage* in the protected form. If later there is a need to access the Passenger location history due to the Passenger's request or other legal ground, *D7. Encrypted Ride Details Storage* can be accessed using the corresponding private key. It is vital that only authorised employees of Bolt Technology OÜ (Controller) have the private key to access the data, for example, to resolve disputes regarding transportation services.

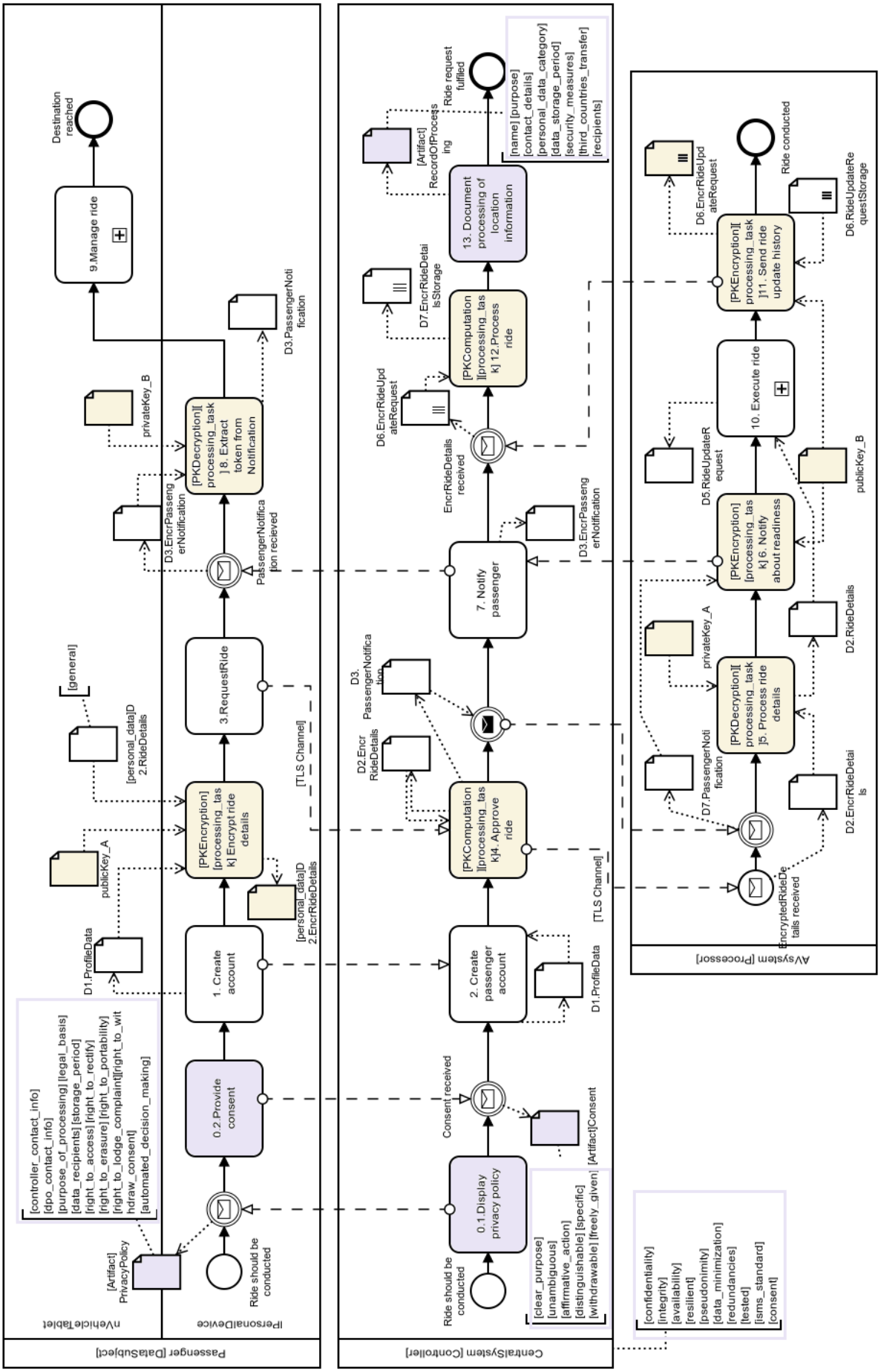

Figure 17. GDPR-compliant Ride Fulfilment process model (PK encryption)

As a result, the demonstrated in Figure 17 business process can be used as a new As-Is process model for the next iteration of analysis using the DPO tool. The GDPR compliance analysis results are shown in Figure 18.

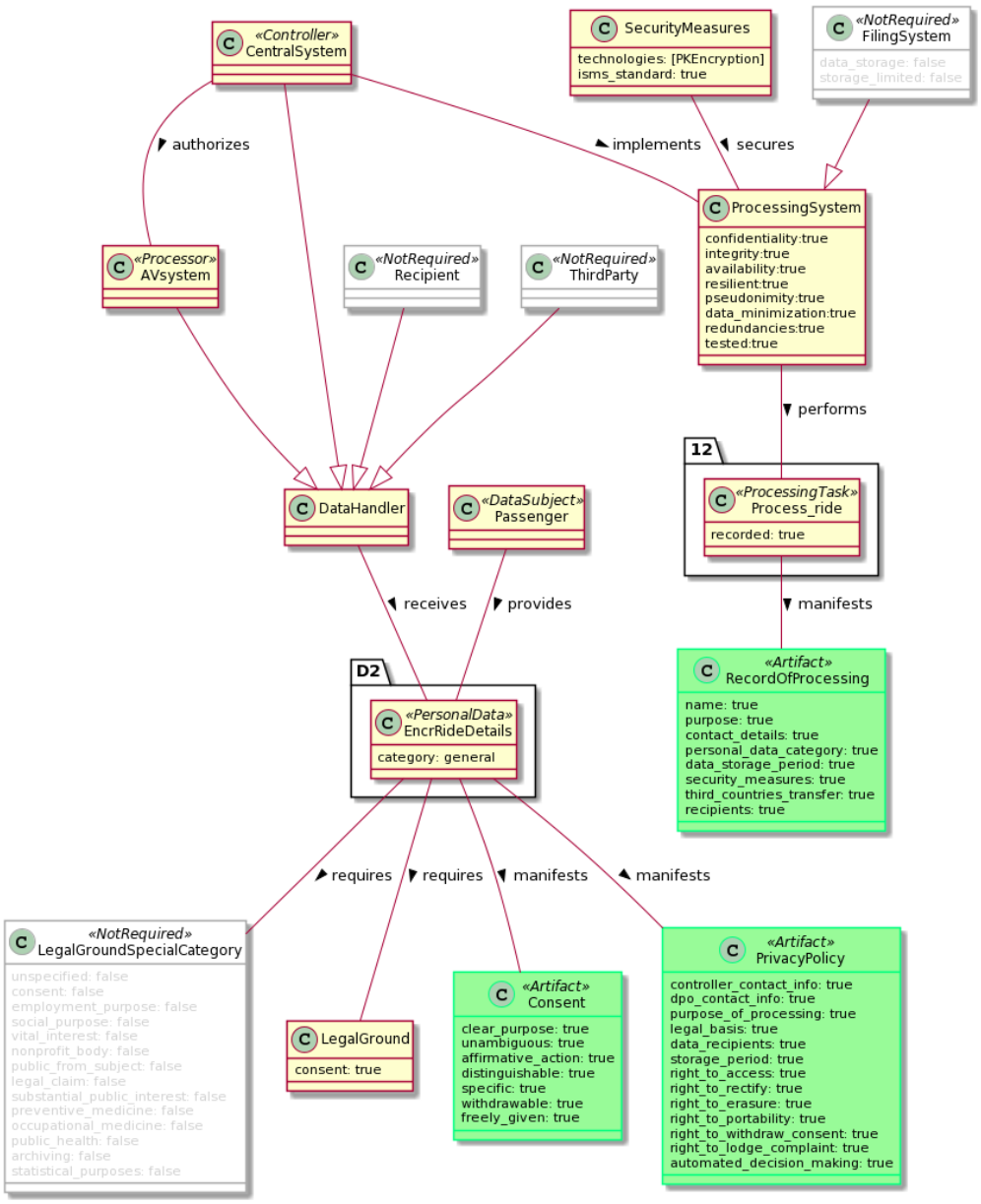

Figure 18. Result of GDPR compliance check: Ride Fulfilment process model (PK encryption)

As we can see, there are no non-compliance issues found in the Design 1 business process model. Therefore, this design can be used by a ride-hailing service provider as a baseline for the Ride Fulfilment process.

Design 2 demonstrates usage of another PET, namely *secret sharing*. According to Pullonen et al. [23], secret sharing is an alternative data protection mechanism to PK encryption. The computation of the protected by secret sharing data is conducted with secure multiparty computation

(MPC) requiring multiple participants. Similarly to PK encryption, secret sharing includes data protection, data computation and opening data confidentiality phases. The initial separation of the data into shares ensures data protection. After, the shares are supplied to parties. Each party retains one private share and executes computation on it to produces output using homomorphic properties of the sharing. To open confidentiality of the protected with secret sharing data, the secret sharing reconstruction should be executed. The reconstruction procedure revert shares into open data if enough shares are obtained.

The implementation of secret sharing within the Ride Fulfilment process is shown in Figure 19, where yellow objects represent technical measure implementation and purple - other changes made in the As-Is model to resolve the non-compliance issues. One of the prerequisites for this design implementation is the possibility to use multiple independent computational servers of the same processing capacity, which conduct computation of the sensitive data in collaboration. One should pay attention that the servers used for the MPC should be independent, meaning that they cannot be both under the management of one business entity or should not cooperate. In the case of Bolt's Ride Fulfilment process, according to to [69], the personal data is stored "in the data centres of Zone Media Ltd. and/or Amazon Web Services, Inc., which are located in the territories of the Member States of the European Union." Thus, Bolt can implement secret sharing using the existing resources, namely using Zone Media Ltd. and/or Amazon Web Services, Inc., for processing and storing shares as one party in addition to the servers where *Central System* runs (owned by Bolt). Additionally, the independence of these processing entities should be guaranteed to prevent possible cooperation and the possibility to get access to protected data by shares reconstruction. It should be noted again that in the context of the Ride Fulfilment process, *Ride Details* and *Ride Update Request* contains sensitive data about Passenger's location and, therefore, need to be protected.

Secret sharing can be used twice during the process: (*i*) for the initial processing of Ride Details to approve a ride; and (*ii*) for Ride Update Request and Ride Details post-processing. In the first case, Passenger should split *Ride Details* into shares to provide them to different servers of *Central System*. Such a split enables separate data into one share, which contains details required for assigning a vehicle to the ride (e.g., current Passenger's location and destination), while another share embraces Passenger's details (e.g., Passenger's name and payment details). Therefore, the first share is used by *Central System Server1* to calculate payment details and Passenger's credentials, and the second share is used by *Central System Server2* to define a vehicle that will execute the ride and Passenger's credentials required for accessing the vehicle. When Passenger obtains both shares, secret sharing reconstruction is executed, so Passenger has all the credentials and details needed for the ride execution. Meanwhile, the second share is also provided to *AV system* which are provided with the shares from the AV system, and as it contains only Passenger's location and part of credentials, *Ride Details* data object is still protected. The second case of secret sharing usage in the process is secret sharing of the Ride Details history. The data protection is implemented by *AV System* during the *11. Finish ride* activity. The separation of Ride Details history ensures that none of the Central System servers has the full history of Passenger's location. However, each server can execute computation on his share to produces output using homomorphic properties, and as a result - adjust maps databases and conduct required learning of algorithms. Thereby, *Central System* server does not have access to the ride logs. Only the Passenger (or administrator on behalf of the Passenger) can reconstruct the full ride details history with the Passenger's location by extracting shares from the Central IS servers that participated in the computation and conducting shares reconstruction.

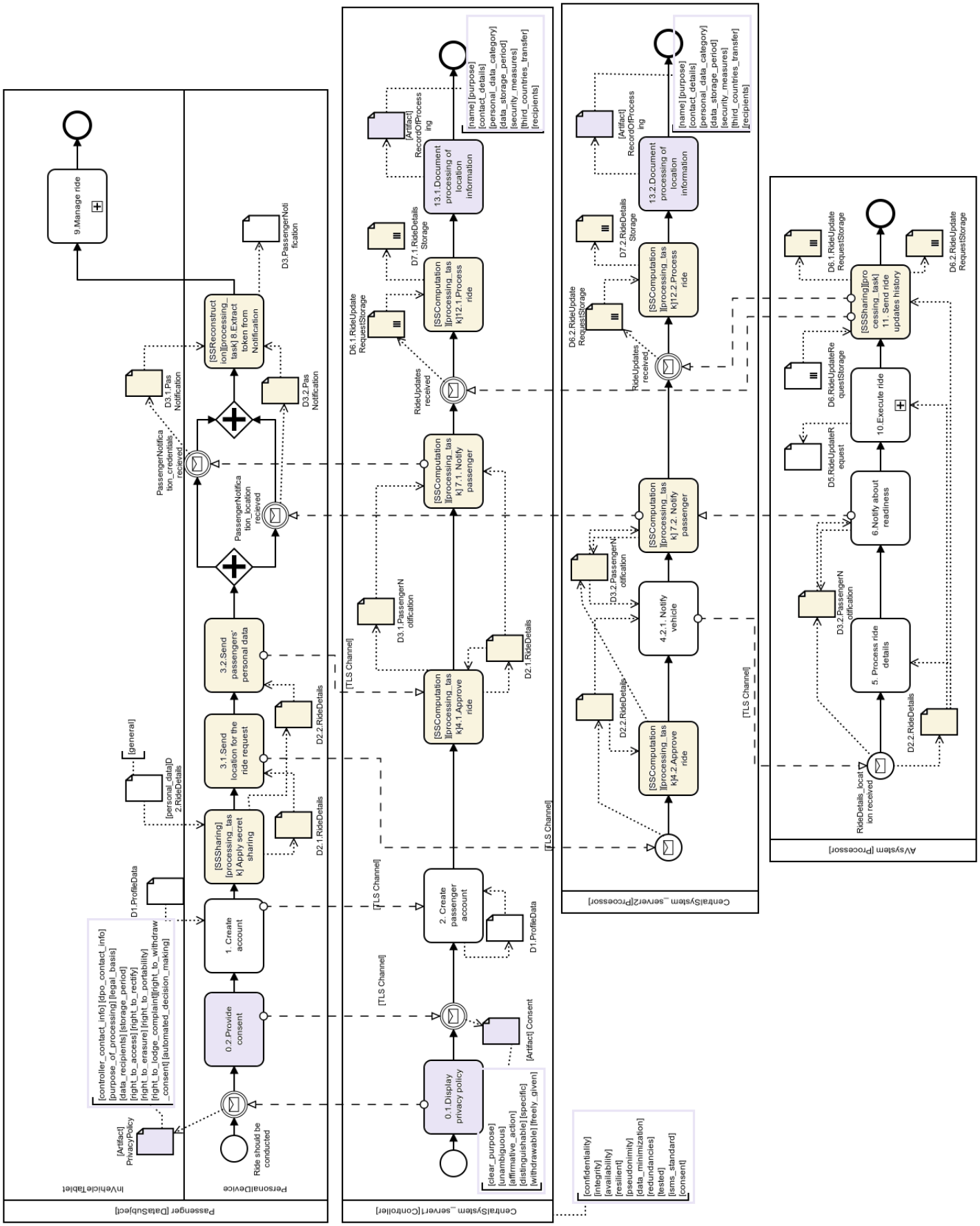

Figure 19. GDPR-compliant Ride Fulfilment process model (Secret Sharing)

As a result, demonstrated in Figure 19 business model can be used as a new As-Is process model for DPO tool analysis. The GDPR compliance analysis results are shown in Figure 20.

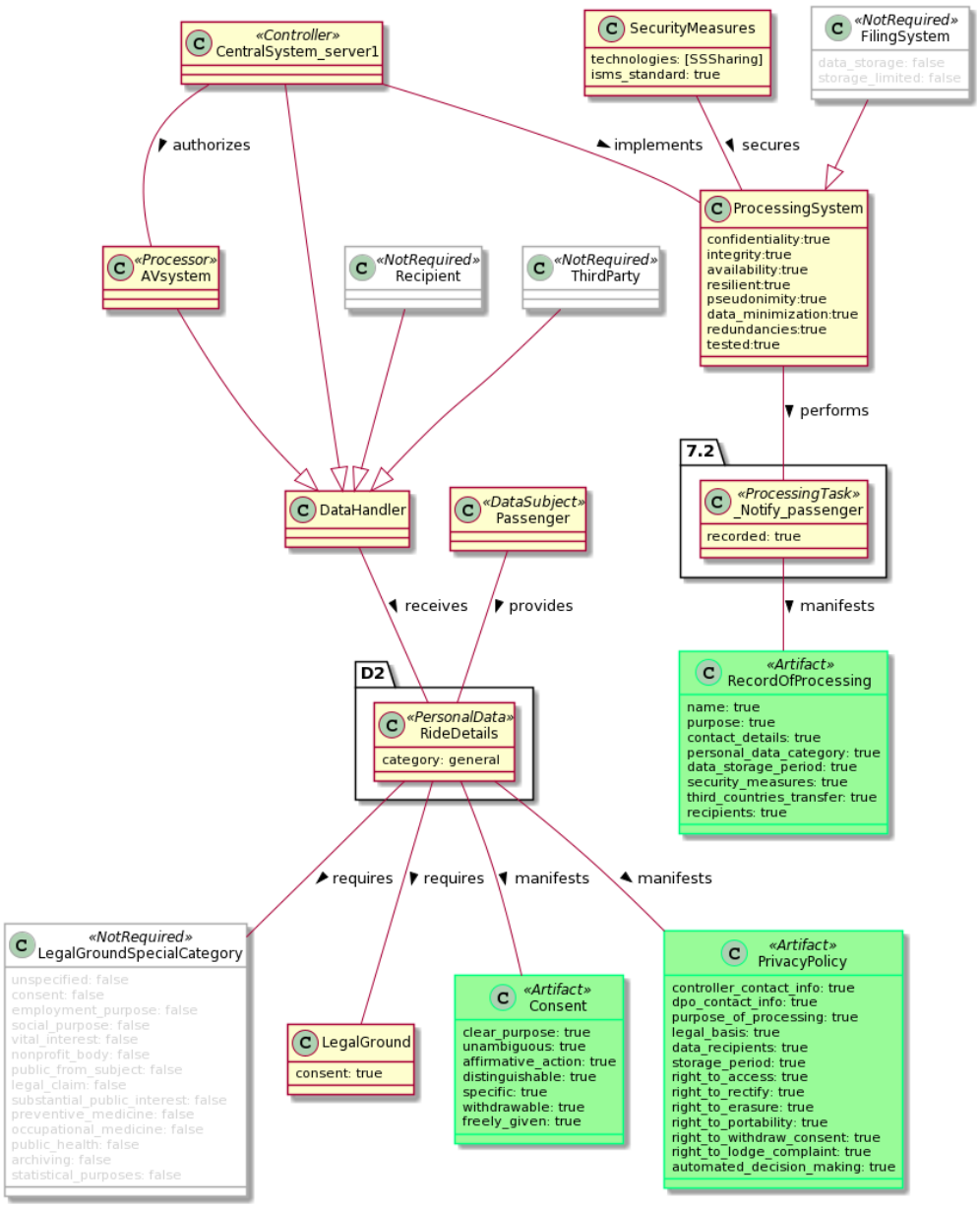

Figure 20. Result of GDPR compliance check: Ride Fulfilment process model (Secret Sharing)

As we can see, there are no non-compliance issues found in the Design 2 business process model. Therefore, this design can be used by a ride-hailing service provider as another baseline for the Ride Fulfilment process.

One important limitation of the GDPR model usage for checking compliance is that it conducts the compliance analysis per task. Thus, resolving the compliance of the process for the task *Notify passenger* or *Process ride* means an extension of the model with additional activities required by the legislation. However, the added activities are considered as sub-activities of the initial one, not

separate ones. Therefore, for example, activities *11.1/11.2. Process ride* should be considered as still one activity *Process ride* presented in Figure 15, but sub-activities of which are executed by separate independent entities. Such representation of the business process activities origins from the initial business tasks of a data controller. Thus, from the perspective of the data controller, activities *11.1/11.2. Process ride* are still one activity, but we separate it for the purpose of ensuring privacy with respect to GDPR by explicitly presenting the used technology implementation (in case of separation of the*Process ride* activity) or by explicitly presenting required sub-activity (e.g., by adding the "Provide consent" sub-activity). Although, such supportive activities as obtaining consent or documenting the processing in the later stages can be considered as processing tasks and, thus, analysed for its compliance to the legislation separately in the later stages of privacy management.

To sum up, we have analysed the modelled business process of Ride Fulfilment in Bolt, the existing privacy policy for passengers and have applied a tool-supported GDPR compliance check. We have demonstrated how identified non-compliance issues can be resolved, providing recommendations to the business process, privacy policy, and consent changes. Also, we have created two designs that illustrate the application of different PETs to the scenario.

4.3 Disclosure Analysis

Once the business process and the system are compliant with the local legislation, we are coming to privacy risks. To secure personal data, a set of PETs that brings more positive impact on personal data protection should be identified. When deciding on the privacy-enhancing technology to use, one needs to consider the level of data visibility the technology delivers and the disadvantages of the PET usage in the context of the business process. The relevance, difficulty to implement, and the created impact should be considered while choosing PETs for the system. Primarily a privacy-enhancing technology should be chosen to minimize risks of personal data leakages. Therefore, having two designs created in Section 4.2.2, we compare them by analyzing PET's effectiveness in terms of personal data disclosure in this subsection. The analysis is conducted employing the Pleak tool[10], described earlier in Section 2.1.

Demonstrated in Figure 21 screenshot of the Pleak contains disclosure analysis results for the As-Is process (see Fig. 15) where no PETs are used for the personal data protection.

Ride Fulfilment Process model (As-Is).bpmn - Simple disclosure analysis report

#	D1. ProfileData	D2. RideDetails	D3. Device Notification	D4. Passenger Notification	D4. Passenger Notification_FIRST	D5. Ride Update Request	D6. Ride Update Request Storage	D7. RideDetails Storage
AVsystem	-	V	O	V	-	V	O	-
CentralSystem	V	V	V	-	O	-	V	V
PersonalDevice	O	O	-	-	V	-	-	-

Figure 21. Disclosure analysis results: Ride Fulfilment process model (As-Is) (V - visible, O - owner, H - hidden)

[10]Pleak Tool can be accessed at `https://pleak.io/` (account: *demo@example.com*, password: *pleakdemo*, manual: `https://pleak.io/wiki/`) [33]

As we can see, all of the shared data is visible to any actor it is sent to. For example, the Central system is an intermediary party that enables Passenger–AV interaction. Therefore, Passenger initially sends its location within the Ride Details data object to the Central system to process the information, assign the ride and transfer Ride Details to the assigned vehicle. Meanwhile, having no data protection mechanism applied to the transferred from Passenger data, the Central system has access to sensitive information. Moreover, at the end of the ride, the Central system receives ride updates and has access to the history of the Passenger's location during the ride. As a result, the Central system becomes a party that has access to any sensitive data of each Passenger that increases the risks of targeting the system with the intention to obtain the personal data. Thereby, disclosure analysis results gives an understanding of data visibility, which sheds light on possible leakage points. Such leakages represents potential vulnerabilities in the systems that should be addressed during the discussed earlier security risk management.

4.3.1 Design 1: Public Key Encryption

The first created design uses one of the most common PETs for protection confidentiality - *Public Key Encryption*. While Figure 17 is focused on meeting the GDPR requirements and enriches the Ride Fulfilment process with activities needed for being the legislation-compliant, Figure 22 illustrates how PK encryption is applied to the Ride Fulfilment process, focusing on the activities and artefacts required for the selected PET implementation. Thus, activities marked with the purple colour in Figure 17 are omitted in Figure 22 as they are considered as supportive sub-activities of the presented tasks. Additionally, the first step *1. Create account* is not considered in the disclosure analysis as there is no processing of information about the Passenger's location. On the model, the red data objects represent the publicly available data, and the green - protected data hidden from access to sensitive information.

On the depicted business process model, a data object *D4. Ride Details* contains sensitive personal data - Passenger location and the current location during the ride conduction. The Passenger owns the data initially in the readable form on his personal device. After applying a *public key A* data object to *D4. Ride Details* during the *Encrypt ride details* activity, the hidden for the reading data object *D4. Encrypted Ride Details* is created. The protected data in the form of a ride request is transmitted to the Central system via a secure channel. Thus, the protected data is processed by the Central system in the *4. Approve ride* activity, exploiting the homomorphic property of the encryption. After that, the protected data is transmitted to the AV system via a secure channel, and the system decrypts the received data using a *private key A* from the keys pair. Thereby, the sensitive data, namely the Passenger location, becomes visible to the AV system. Afterwards, when the ride has been executed, the system has another data artefact to be protected - *D6. Ride Update Request*. The artefact is created as a result of generated by the Passenger requests for the ride changes, including the change of destination and routes, and therefore, contains the history of the Passenger's location during the whole ride. To protect the sensitive data, the AV system encrypts its storage during the *11. Protect ride update storage* activity. The protected data is transmitted to the Central System via a secure channel. The Central System processes it using the PK computation in order to improve future services. Finally, the protected ride details history is stored in the storage so that the authorized person can access it (e.g. Passenger or authorized employee) using the *private key B* for decryption.

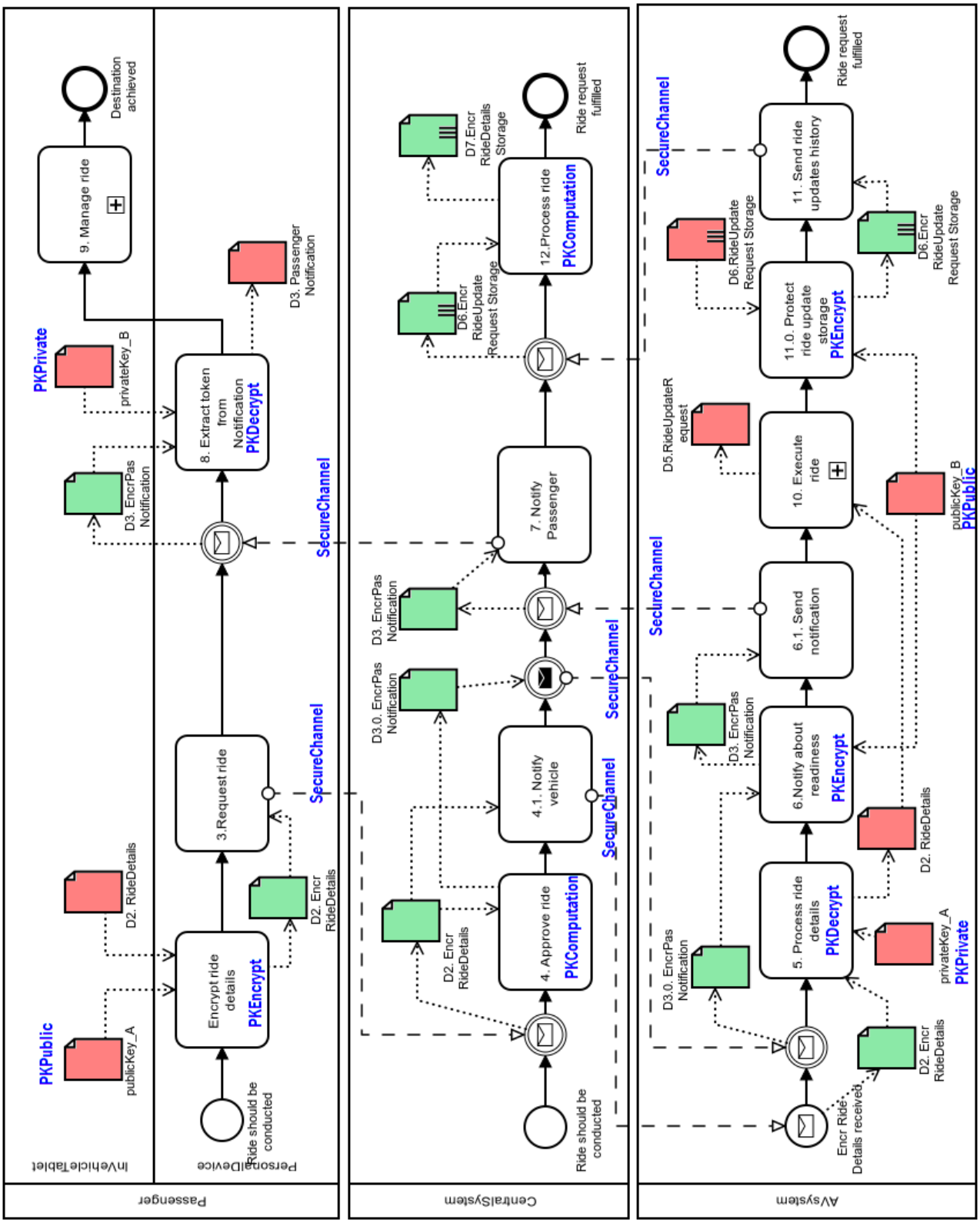

Figure 22. Ride Fulfilment process model (PK encryption)

Now, having a PE-BPMN model that depicts created Design 1, we can employ Pleak tool. The results of disclosure analysis from the tool is presented in Figure 23.

Ride Fulfilment process PK encryption.bpmn - Simple disclosure analysis report

#	D2. Encr RideDetails, D2. RideDetails	D3. EncrPas Notification, D3. Passenger Notification, D3.0. EncrPas Notification	D5.RideUpdateRequest	D6.Encr RideUpdate Request Storage, D6.RideUpdate Request Storage	D7.Encr RideDetails Storage	privateKey_A	privateKey_B	publicKey_A	publicKey_B
AVsystem	V	H	V	O	-	O	-	-	O
CentralSystem	H	H	-	H	H	-	-	-	-
PersonalDevice	O	V	-	-	-	-	O	O	-

Figure 23. Disclosure analysis results: Ride Fulfilment process model (PK encryption) (V - visible, O - owner, H - hidden)

4.3.2 Design 2: Secret Sharing

Another way of protecting personal data is its separation into shares which are shared to different processors. The application of *secret sharing* in the Passenger–AV interaction is illustrated above in Design 2 of the Ride-hailing Fulfilment process. Figure 24 shows the PET usage focusing on the sensitive data disclosure. Similarly to the previous design analysis, Figure 19 differs from Figure 24 as the latter illustrates how secret sharing is applied to the Ride Fulfilment process, focusing on the activities needed for the selected PET implementation. Thus, activities marked with the purple colour in Figure 19 are omitted in Figure 24 as they are considered as supportive sub-activities of the presented tasks. Additionally, the first step *1. Create account* is not considered in the disclosure analysis as there is no processing of information about the Passenger's location.

On the depicted business process model, a data object *D2. Ride Details* contains sensitive personal data - Passenger location and the current location during the ride conduction. The Passenger owns the data initially in the readable form on his device. After executing secret sharing by dividing the data object into shares, it is hidden for the reading data objects *D2.1. Ride Details* and *D2.2. Ride Details* are created. The first share should contain only the part of the ride details object required for processing Passenger's details (e.g., the account details, payment details) and notifying them during the ride Fulfilment. The second share should contain only data required for assigning a vehicle to the ride. The created shares are transmitted to the Central system servers via secure channels. Then, both servers execute secret sharing computation to approve the ride and create two shares of the *Passenger Notification* (D4.1 and D4.2), using the received shared of the ride details. The created Passenger Notification shares contain either passenger personal data specific notification that should be transmitted directly to the Passenger (e.g., payment details) or the vehicle-related information (e.g., credentials for Passenger's verification) that should be transmitted to the assigned vehicle. It should be noted that each Passenger Notification share separately does not give access to the sensitive personal data. Thereby, the personal data is safe.

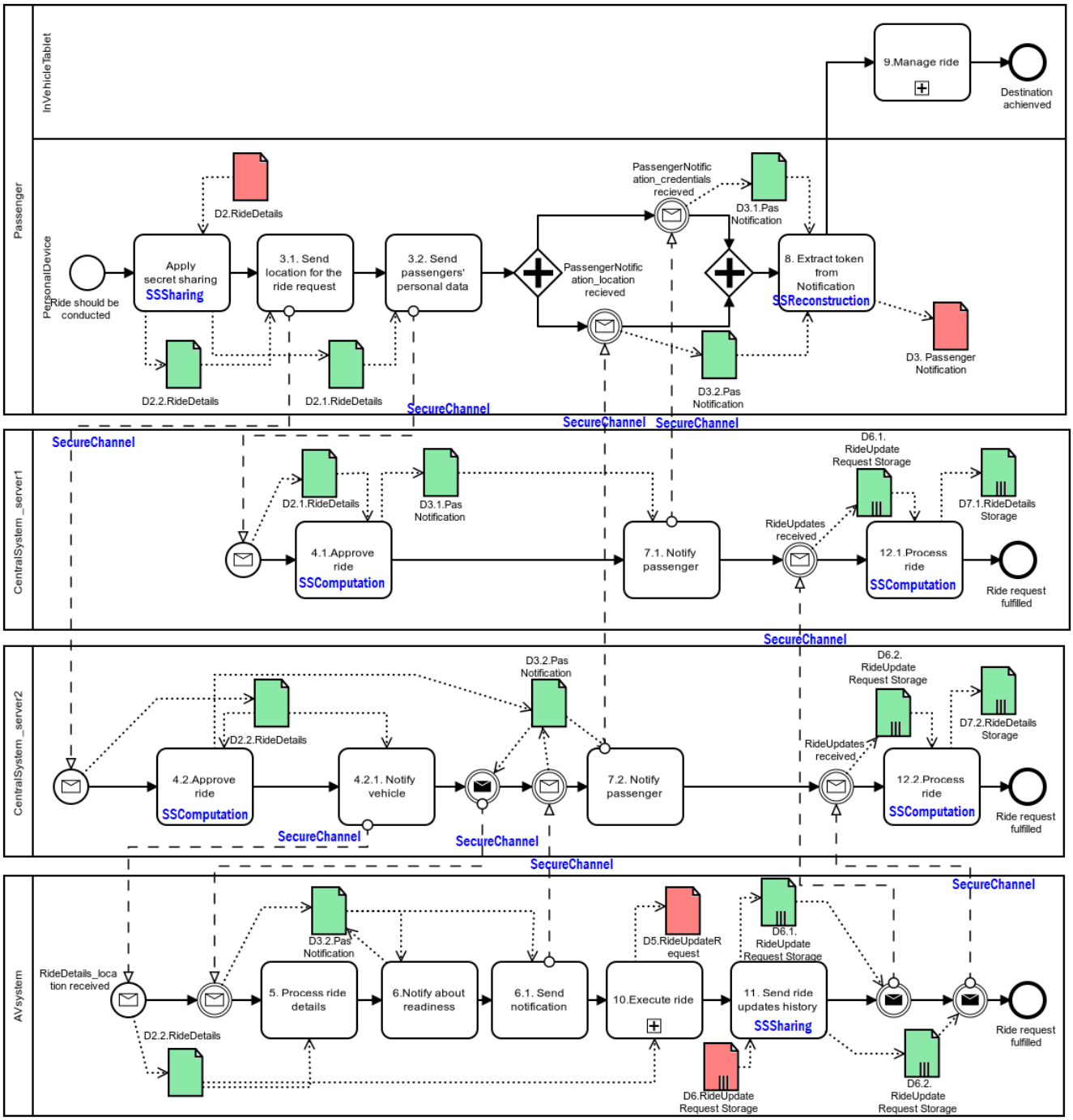

Figure 24. Ride Fulfilment process model (Secret Sharing)

After the first server of the Central system finishes the riding assignment, it transfers the created Passenger Notification share to Passenger. Once the vehicle gets to the starting point of the ride, it notifies the Central system server from which the share was received (*Central system server 2*) about readiness to start the ride and sending back the *Passenger Notification* share securely. As a result, the central system (server 2) sends its share of Passenger Notification to Passenger. As soon as Passenger has received two shares of *Passenger Notification*, secret sharing reconstruction can be executed. Consequently, Passenger obtains access to D4. Passenger Notification and can manage the ride.

Passenger can modify the initially selected route, destination, or the driving mode during the ride by creating *Ride Updates*. After the Passenger is taken to the destination, all the created during the travel updates form the artefact *6. Ride Update Request Storage* which stores the entire history of the Passenger's location during the ride. As it is sensitive personal data, such a location history should be protected. Therefore, the secret-sharing PET can be applied to the data, creating two shares of the history data object. The shares should be transmitted to different Central system servers via the secure channels, and after the execution of secret sharing computation for the internal algorithm and map updates, the location history shares are stored. Due to the separation of location history, none of the Central system servers can access the entire history that assures sensitive data protection. Finally, the authorized person can access the protected history (e.g., Passenger or authorized employee) using the secret-sharing reconstruction.

The disclosure analysis with Pleak enables to identify the level of data objects visibility to the process participants. The analysis results are presented in Figure 25.

Ride Fulfilment process Secret Sharing.bpmn - Simple disclosure analysis report

#	D2.1.RideDetails, D2.2.RideDetails, D2.RideDetails	D3. Passenger Notification, D3.1.Pas Notification, D3.2.Pas Notification	D5.RideUpdateRequest	D6.1. RideUpdate Request Storage, D6.2. RideUpdate Request Storage, D6.RideUpdate Request Storage	D7.1.RideDetails Storage, D7.2.RideDetails Storage
AVsystem	H	H	V	O	-
Central System _server1	H	H	-	H	H
Central System _server2	H	H	-	H	H
PersonalDevice	O	V	-	-	-

Figure 25. Disclosure analysis results: Ride Fulfilment process model (Secret Sharing) (V - visible, O - owner, H - hidden)

4.3.3 Comparison of Designs

Having the results of the disclosure analysis, we can compare the levels of data protection and visibility to the process participant delivered with the selected PETs.

Through protecting sensitive data with secret sharing, Central System servers cannot see any data, and the system only plays a role of an intermediary. Meanwhile, only Passenger and AV system can read the sensitive data as the former is its owner, and the latter needs to see the passenger's location for executing the ride.

As can be seen, both PK encryption and secret sharing enables the same level of visibility of *Passenger Notifications*, *Ride Updates*, and *Ride Details Storage*. However, secret sharing enables making *Ride Details* hidden from the AV system, while using PK encryption provides the full

visibility of the Ride Details to the vehicle. Meanwhile, the usage of PK encryption for Ride Details requires passenger's Personal Device to conduct extra activities, namely, secret sharing of Ride Details and further Passenger Notification shares reconstruction. Therefore, there should be set prerequisites for processing capabilities of Personal Device; otherwise, the Ride Fulfilment processing time may radically increase. We do not recommend adding complexity and processing time to the Ride Fulfilment by secret sharing usage for Ride Details. Almost the same level of sensitive data protection can be delivered by PK encryption if the Central System sends to AV system only the minimum required to vehicle part of Ride Details, creating it through PK computation.

Regarding the second illustrated case of secret sharing applied to *Ride Update*, it can be seen that both PETs ensures the same level of Ride Update and Ride Details Storage visibility. However, Encrypted Ride Details Storage can be accessed by an adversary easier than stored in a distributed manner with secret sharing, as it requires only have private key and access to the storage itself. Meanwhile, for secret shares reconstruction, an adversary has to obtain access to both Central System servers to obtain two shares and conduct identity spoofing of the authorized stakeholder who can conduct shares reconstruction. Thereby, the likelihood of getting access to the Ride Details Storage is higher for PK encrypted protection, and as a result, secret sharing is preferred. Moreover, in the case of Bolt's Ride Fulfilment process, according to to [69], the personal data is stored "in the data centres of Zone Media Ltd. and/or Amazon Web Services, Inc., which are located in the territories of the Member States of the European Union." Thus, Bolt can implement secret sharing using the existing resources, namely using Zone Media Ltd. and Amazon Web Services, Inc., for storing Ride Details history share. The final proposed design is depicted in Figure 26.

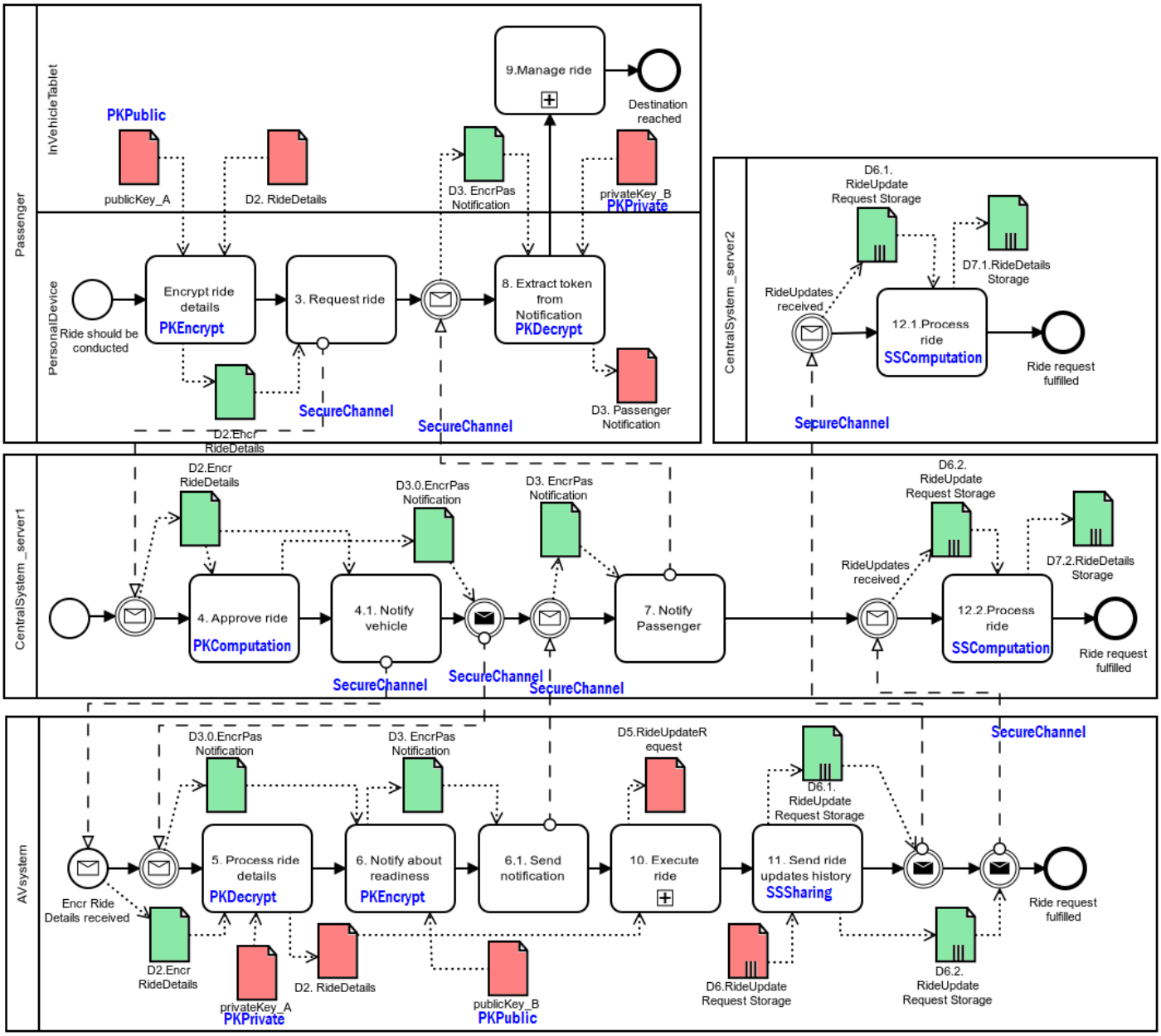

Figure 26. Ride Fulfilment process model: proposed protection

Figure 27 contains the results of disclosure of analysis for the proposed final design.

Ride Fulfilment process Secret Sharing.bpmn - Simple disclosure analysis report

#	D2.1.RideDetails, D2.2.RideDetails, D2.RideDetails	D3. Passenger Notification, D3.1.Pas Notification, D3.2.Pas Notification	D5.RideUpdateRequest	D6.1. RideUpdate Request Storage, D6.2. RideUpdate Request Storage, D6.RideUpdate Request Storage	D7.1.RideDetails Storage, D7.2.RideDetails Storage
AVsystem	H	H	V	O	-
Central System _server1	H	H	-	H	H
Central System _server2	H	H	-	H	H
PersonalDevice	O	V	-	-	-

Figure 27. Disclosure analysis results: Ride Fulfilment process model (proposed protection)

Thereby, Central System as an intermediary for Passenger–AV interaction has no access to the sensitive data, and only the Passenger and the assigned vehicle can see the personal information.

4.4 Summary

In this chapter, managing the privacy of personal data is discussed answering *RQ2*. Primarily, the researched case is checked on being compliant with the local personal data protection legislation - GDPR. After, to comply with the regulation, the researched process are analysed to define how to implement security measures for enhancing personal data privacy. Therefore, two designs that implement the privacy-enhancing technologies - public key encryption and secret sharing - are created. Next, the created designs are analysed on the data disclosure. Finally, the Ride Fulfilment Process that embraces public key encryption and secret sharing is proposed.

For addressing the personal data management in the Ride Fulfilment scenario, we conduct a tool-supported analysis using the DPO tool (for conducting the GDPR-compliance check) and the Pleak tool (for conducting disclosure analysis). The tools can be used by the controller together or separately, depending on the researched scope. The DPO tool is suitable for the data protection officer's usage when analysing organisation processes on the data protection rules compliance. The tool enables a high-level analysis of the process. Meanwhile, the Pleak tool enables analysis of the privacy assurance on a lower implementation level. Thus Pleak is suitable for usage by both security and data protection specialists when analysing the delivered data protection level of the specific privacy-enhancing technologies. The used tools are open source so that they can be freely accessed and used by anyone. However, both tools are prototypes and, therefore, have limited user functionality (e.g., the created models and analysis reports should be stored locally on users computer as there is no long-term storage inside tools).

5 Conclusion

The current thesis is based on the research activities conducted within the autonomous driving lab. The thesis's main goal is to determine how the information in the Passenger–Autonomous Vehicle interaction can be protected. Therefore, we discuss how to assure data protection considering two approaches - information privacy and security risk management. To address security risk management we use a threat-driven approach for the security requirements elicitation. As a result, we have elicited a set of security requirements and recommended how to make its validation and prioritisation before integrating into systems. As the Ride Fulfilment process includes processing personal passenger data, specific methods for protecting personal data privacy are discussed. Finally, the tool-supported analysis of the personal data management within the business processes are conducted, which results in the proposed design of the privacy-enhancing technologies usage for the researched case.

5.1 Answers to Research Questions

The thesis considers ***how the information in Passenger-AV interaction can be protected.*** The research is divided into two parts that address the main research goal from different perspectives. In this section, answers to the stated research sub-questions are given.

RQ1: How can information security risks in Passenger-AV interaction be managed?

To manage information security in the Passenger–AV interaction scenario, security risk management can be applied. Information security risks are threat-driven. Thus we have defined assets that can be targeted and the threats which may cause harm to the assets. In the context of an autonomous vehicle that conducts the ride, the protected assets are components of the information system that manage the vehicle and the ride-hailing service. The assets include both business and system ones. The business assets include data entities that contain data required for the ride preparation and execution. System assets in the designed case are presented by components of the AV system's application layer and the component of the ride-hailing service provider system that initially enables the ride ordering. In the researched case, 22 security risks are defined based on the threat model determined previously from the threat libraries and taxonomies considering the protected assets. We observes that most of the risks are targeting the *Passenger Notification* asset that contains the passenger's current location, while accessing the *Ride Details* assets that embrace the ride information and coordinate the ride execution may cause crucial harm for the ride safety and passenger's privacy. Finally, the security requirements that mitigated the identified risks are proposed. As a result, we have defined the primary attack vectors for the Passenger–AV interaction and a set of security measures to increase the level of information security is proposed. The described process of addressing the information security within the Bolt AV system can be used as a baseline framework for security risk management within other ride-hailing systems that implement Passenger–AV interaction in a similar way.

RQ2: How can the privacy of Passenger's personal data be managed in the Passenger-AV interaction?

To ensure Passenger's personal data privacy, the business process of Ride Fulfilment and the used systems should be compliant with the local data protection legislation. In the case of Ride Fulfilment in Bolt, which operates within the EU countries, the data protection is regulated by General Data Protection Regulation (GDPR). The business process can be analysed manually, or tool-supported

analysis can be done. In the thesis, we demonstrate the DPO tool usage for discovering GDPR non-compliance issues based on the provided business process model. The GDPR-compliance analysis results helps to redesign the As-Is process by highlighting non-compliance issues. Additionally, GDPR requires to employ privacy by design principles and privacy-enhancing technologies. Many different PETs target the same security goals. For defining which PET is better to use, the delivered personal data protection levels can be compared using the tool-supported disclosure analysis. The Pleak tool identifies the data visibility levels within the process that helps to detect possible data leakages. As a result, the tool helps compare PETs and create the business process design that ensures the required personal data protection level. A few process designs that are compliant with the GDPR requirements are proposed and the possible data leakages are pinpointed based on the data disclosure analysis results. To sum up, for assuring personal data protection in the Passenger–AV interaction, the service provider systems should employ privacy by design principles, comply with the local data protection legislation, use privacy-enhancing technologies besides of ensuring information security. Specialised business process analysis tools can be utilised to help with the existing interaction process analysis.

5.2 Lessons Learnt

To address the case analysis's primary goal, we have conducted the literature review to find the methodology of information risk analysis for Passenger–AV interaction. We have discovered a knowledge gap in the method of assuring personal data protection in the autonomous vehicles field. Also, we have found that to guarantee personal data protection, there are two different approaches - from the perspective of privacy protection and information security, while the latter is a required criterion of ensured former.

We have applied the threat-driven approach to security requirements elicitation for Passenger–AV interaction. The research results in the demonstrated approach of security requirements elicitation based on the created threat model. It should be noted that the threat model contains threats that are applicable for the Passenger–AV interaction in the context of the ride-hailing service, where an external provider delivers an infotainment system, meaning that the system is not integrated into the AV. For the further usage of the presented framework, the system architecture should be considered, as based on it, the threats may result in risks different from the provided in the case analysis.

The presented threat model and approach of ISSRM aim to address only risks related to the intentional harmful attacks, leaving out of scope unintentional threats caused by the system users. Furthermore, for applying the presented result to other cases, one should pay attention to the used stack of technologies in the system implementation, as there may exist several technology-specific attacks, and thus, they are not relevant to the purpose of our research.

Considering personal data management, the AV system developers, ride-hailing service providers, and risk managers of both organisations should pay attention to the local legislation regulating personal data processing. Also, while the legislation compliance and the privacy-enhancing technologies influence can be analysed manually using internal organisation's surveys and templates, the tool-supported analysis can increase accuracy and decrease time consumption compared to the manual assessment [31].

It is beneficial to employ security by design and privacy by design approaches to deliver the developed systems as they are already not just recommended but required by some legislation (e.g., GDPR). Also, choosing PETs for the developed system, a system analyst should consider a

combination of few different PETs throughout the system considering the applicability, cost-effective trade-off, and the delivered data protection.

The research has been conducted using the case study. This method in the autonomous vehicle field is especially prominent and applicable, as the field is relatively new and the technology is still under development. Consequently, the phenomenon is not yet determined, and the approaches for tackling it are not standardised either. Therefore, to check the applicability of the existing methods in the autonomous vehicle context, a case analysis is a fruitful method for discovering valuable insights into current theories usage and broad field. Moreover, this case study is beneficial for the infrastructure and process in Bolt Technology OÜ used in the case study, but the study results are also supposed to motivate further research of information security management.

The thesis scope is limited by Passenger–AV interaction and focused on the application layer of the AV system. However, for ensuring data protection in autonomous vehicles, similar research should be conducted to analyse information security risks and personal data usage in all the other processes in AV (e.g., vehicle-to-infrastructure and vehicle-to-vehicle interactions).

5.3 Limitations

The system developers confirm that the received results are relevant, valid, and help to secure the Passenger–AV interaction. However, further validation of the derived security requirement is needed to evaluate its coverage. We do not declare that the created threat and countermeasure models are complete, but we aim to propose them as a baseline for a systematic approach for improving autonomous vehicles' system quality for the ride-hailing service providers.

As the thesis covers the case-specific analysis, the personal data management recommendations are based on the local legislation - namely, GDPR - the primary regulation in EU counties. Hence, the employed DPO tool is not applicable for analysing a Ride Fulfilment process outside the EU. Besides, it should be noted that the selected for comparison PETs - public key encryption and secret sharing - are selected due to their wide usage, ease to illustrate and relevance for Bolt's system. However, to assure personal data protection within other service providers' systems, alternative privacy-enhancing technologies should also be considered.

The further limitations of the current research are based on the used method - case study. Firstly, the studied Passenger-AV interaction scenario is company-specific, and therefore, the obtained results may be biased. This bias can be for the threats in Bolt's systems' concrete case, requirements evaluation, and the privacy-enhancing technologies implementation. Another limitation is the low possibility to generalize the results without conducting further research within other AV systems development projects. Mostly, this limitation is relevant for the created threat model, as it is asset-based. As different system implementations tend to have different vulnerabilities, consequently - threats, the defined threat model may need further development for its usage within other AV systems management.

5.4 Future Work

Our study researches the complex system consisting of an autonomous vehicle, an external service provider, and a human (a passenger in the researched case). Thus the risk mitigation measures for treating security risks include not only system requirements. One should also assess internal security policies, user guidelines and introduce changes for meeting security criteria. Further research should

be done to extend our result with attacks assessment, including the likelihood and impact of the threats and overall assessment of the risk level. Additionally, evaluation of the derived security requirements elicitation framework is needed. The evaluation can be done by comparing the derived security requirements set and its coverage.

In future work, the presented results, namely the threat and risk models, also can be generalised by applying them to another case of Passenger–AV interaction, for example, in the case of the other system architecture. Also, we strive to raise the discussion about the importance of implementing the security requirements for the Passenger–AV interaction for treating security and privacy issues. The ensured secure Passenger–AV interaction would increase society's trust in general and single users to autonomous systems and ensure a high-quality user experience.

The thesis demonstrates that addressing passenger's data protection should be done from two different analysis viewpoints - privacy management and security risk management. The future work potentially is to understand the dependencies and correlation of both approaches – how one depends on another and how one helps another.

During research in the thesis, some broader directions for future work have been defined. Firstly, the research of Passenger–AV interaction can be expended for the cases of few passengers who participate in a ride. Secondly, as intelligent transportation systems (ITS) are getting more and more developed, the context of the research interaction changes. Therefore, the information security of Passenger–AV interaction in the environment of connected vehicles and ITS should be analysed. Finally, while the internet of things (IoT) is more used in everyday life, the connection of personal smart gadgets with autonomous vehicles may bring a more seamless experience of service usage. For example, confirmation of the ride or passenger's verification may be used using personal smart gadgets. However, an increase in the complexity and interconnections of the systems enables new security attacks. Therefore, information security and personal data management in autonomous vehicles should be research in the context of multiple systems connections (e.g., ITS and IoT) and multiple users. Continuing the research in the mentioned above direction will significantly improve the autonomous vehicle systems' overall security and benefit to creating convenient and safe intelligent transport infrastructure.

References

[1] M. A. Assaad, R. Talj, and A. Charara, "Autonomous driving as system of systems: roadmap for accelerating development," in *2019 14th Annual Conference System of Systems Engineering (SoSE)*, pp. 102–107, 2019.

[2] SAE international, "Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles," *SAE International,(J3016)*, 2016.

[3] F. Jahan, W. Sun, Q. Niyaz, and M. Alam, "Security modeling of autonomous systems: A survey," *ACM Comput. Surv.*, vol. 52, Sept. 2019.

[4] M. Hagenzieker, R. Boersma, P. N. Velasco, M. Ozturker, I. Zubin, and D. Heikoop, "Automated buses in europe: An inventory of pilots," tech. rep., TUDelft, Feb. 2020. Version 0.5.

[5] "Driverless public bus route now open in Tallinn." `https://e-estonia.com/driverless-public-bus-tallinn/`. Last accessed 28-Oct-2020.

[6] European Union Agency for Cybersecurity, "ENISA good practices for security of Smart Cars." `https://www.enisa.europa.eu/publications/smart-cars`, November 2019. Last accessed 14-Oct-2020.

[7] X. Sun, H. Chen, J. Shi, W. Guo, and J. Li, "From HMI to HRI: Human-Vehicle Interaction Design for Smart Cockpit," in *Human-Computer Interaction. Interaction in Context* (M. Kurosu, ed.), (Cham), pp. 440–454, Springer International Publishing, 2018.

[8] "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)." `http://data.europa.eu/eli/reg/2016/679/2016-05-04`, 2016. Last accessed 24-Nov-2020.

[9] State of California - Department of Justice - Office of the Attorney General, "California Consumer Privacy Act (CCPA)." `https://oag.ca.gov/privacy/ccpa`, 2018. Last accessed 12-Jan-2021.

[10] IPA information technology-promotion agency, "Approaches for vehicle information security." `https://www.ipa.go.jp/files/000033402.pdf`, August 2013. Last accessed 14-Oct-2020.

[11] C. Hodge, K. Hauck, S. Gupta, and J. C. Bennett, "Vehicle cybersecurity threats and mitigation approaches," tech. rep., National Renewable Energy Lab.(NREL), Golden, CO (United States), 2019.

[12] A. Shostack, *Threat Modeling: Designing for Security*. Wiley Publishing, 1st ed., 2014.

[13] M. Muckin and S. C. Fitch, "A threat-driven approach to cyber security," *Lockheed Martin Corporation*, 2014.

[14] A.-A. O. Affia, R. Matulevičius, and A. Nolte, "Security risk management in e-commerce systems: A threat-driven approach," *Baltic Journal of Modern Computing*, vol. 8, no. 2, pp. 213–240, 2020.

[15] European Union Agency for Cybersecurity, "ENISA: Guidelines for SMEs on the security of personal data processing." https://op.europa.eu/s/ou6H, January 2017. Last accessed 4-Dec-2020.

[16] "Personal Data Protection Act 2012 (No. 26 of 2012)." https://sso.agc.gov.sg/Act/PDPA2012, 2012. Last accessed 12-Jan-2021.

[17] F. Gerry QC, J. Muraszkiewicz, and N. Vavoula, "The role of technology in the fight against human trafficking: Reflections on privacy and data protection concerns," *Computer Law & Security Review*, vol. 32, no. 2, pp. 205–217, 2016.

[18] European Union, "Regulations, Directives and other acts." https://europa.eu/european-union/law/legal-acts_en. Last accessed 16-Dec-2020.

[19] OneTrust DataGuidance, "Comparing privacy laws: GDPR v. Singapore's PDPA." https://www.dataguidance.com/resource/comparing-privacy-laws-gdpr-v-singapores-pdpa, July 2020. Last accessed 12-Jan-2021.

[20] OneTrust DataGuidance, "Comparing privacy laws: GDPR v. CCPA." https://www.dataguidance.com/sites/default/files/ccpa_v_gdpr_latest_edition.pdf, December 2019. Last accessed 12-Jan-2021.

[21] J.-H. Hoepman, "Privacy design strategies," *IFIP Advances in Information and Communication Technology*, vol. 428, pp. 446–459, 2014.

[22] D. Le Métayer, J.-H. Hoepman, S. Schiffner, G. Danezis, M. Hansen, R. Tirtea, and J. Domingo-Ferrer, "Privacy and Data Protection by Design - from policy to engineering," tech. rep., European Union Agency for Network and Information Security (ENISA), January 2015.

[23] P. Pullonen, J. Tom, R. Matulevičius, and A. Toots, "Privacy-enhanced BPMN: enabling data privacy analysis in business processes models," *Software and Systems Modeling*, vol. 18, pp. 3235–3264, Dec 2019.

[24] P. Pullonen, R. Matulevičius, and D. Bogdanov, "Pe-bpmn: Privacy-enhanced business process model and notation," in *Business Process Management* (J. Carmona, G. Engels, and A. Kumar, eds.), (Cham), pp. 40–56, Springer International Publishing, 2017.

[25] A. J. Menezes, S. A. Vanstone, and P. C. v. Oorschot, *Handbook of Applied Cryptography*. USA: CRC Press, Inc., 1st ed., 1996.

[26] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC '09, (New York, NY, USA), p. 169–178, Association for Computing Machinery, 2009.

[27] P. A. Bonatti, S. Kirrane, I. M. Petrova, and L. Sauro, "Machine understandable policies and gdpr compliance checking," *KI - Kunstliche Intelligenz*, vol. 34, no. 3, pp. 303–315, 2020. Cited By :1.

[28] R. Matulevičius, J. Tom, K. Kala, and E. Sing, "A method for managing GDPR compliance in business processes," in *Advanced Information Systems Engineering* (N. Herbaut and M. La Rosa, eds.), (Cham), pp. 100–112, Springer International Publishing, 2020.

[29] J. Tom, E. Sing, and R. Matulevičius, "Conceptual representation of the GDPR: Model and application directions," in *Perspectives in Business Informatics Research* (J. Zdravkovic, J. Grabis, S. Nurcan, and J. Stirna, eds.), (Cham), pp. 18–28, Springer International Publishing, 2018.

[30] K. Kala, *Refinement of the general data protection regulation (GDPR) model: administrative fines perspective*. PhD thesis, Master's thesis, University of Tartu, 2019.

[31] E. Sing, R. Matulevičius, and T. Jake, "A meta-model driven method for establishing business process compliance to gdpr," Master's thesis, University of Tartu, Institute of Computer Science, 2018.

[32] R. Matulevičius, J. Tom, K. Kala, and E. Sing, "A method for managing gdpr compliance in business processes," in *Advanced Information Systems Engineering* (N. Herbaut and M. La Rosa, eds.), (Cham), pp. 100–112, Springer International Publishing, 2020.

[33] A. Toots, R. Tuuling, M. Yerokhin, M. Dumas, L. García-Bañuelos, P. Laud, R. Matulevičius, A. Pankova, M. Pettai, P. Pullonen, and J. Tom, "Business process privacy analysis in pleak," in *Fundamental Approaches to Software Engineering* (R. Hähnle and W. van der Aalst, eds.), (Cham), pp. 306–312, Springer International Publishing, 2019.

[34] R. Matulevičius, *Fundamentals of Secure System Modelling*. Springer, 2017.

[35] M. Whitman and H. Mattord, *Principles of Information Security*. Cengage Learning, 4th ed., 2012.

[36] "ISO/IEC 27001:2013: Information technology — Security techniques — Information security management systems — Requirements." `https://www.iso.org/standard/54534.html`. Last accessed 15-Nov-2020.

[37] "ISO/IEC 27005:2018: Information technology — Security techniques — Information security risk managements." `https://www.iso.org/standard/75281.html`. Last accessed 24-Mar-2021.

[38] "ISO/IEC 27031:2011: Information technology — Security techniques — Guidelines for information and communication technology readiness for business continuity." `https://www.iso.org/standard/44374.html`. Last accessed 24-Mar-2021.

[39] J. T. F. T. Initiative, "Managing information security risk: Organization, mission, and information system view," Tech. Rep. NIST Special Publication (SP) 800-39, Gaithersburg, MD, USA, 2011.

[40] P. Grassi, J. Fenton, E. Newton, R. Perlner, A. Regenscheid, W. Burr, J. Richer, N. Lefkovitz, J. Danker, Y.-Y. Choong, K. Greene, and M. Theofanos, "Digital identity guidelines: Authentication and lifecycle management," Tech. Rep. Special Publication (NIST SP) - 800-63B, 2020.

[41] É. Dubois, P. Heymans, N. Mayer, and R. Matulevičius, *A Systematic Approach to Define the Domain of Information System Security Risk Management*, pp. 289–306. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010.

[42] D. Ganji, C. Kalloniatis, H. Mouratidis, and S. Malekshahi Gheytassi, "Approaches to develop and implement iso/iec 27001 standard - information security management systems: A systematic literature review," vol. 12, pp. 228–238, 12 2019.

[43] W. Abbass, A. Baina, and M. Bellafkih, "Survey on information system security risk management alignment," in *2016 International Conference on Information Technology for Organizations Development (IT4OD)*, pp. 1–6, 2016.

[44] R. Caralli, J. Stevens, L. Young, and W. Wilson, "Introducing OCTAVE Allegro: Improving the Information Security Risk Assessment Process," Tech. Rep. CMU/SEI-2007-TR-012, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, 2007.

[45] C. McCarthy and K. Harnett, "National Institute of Standards and Technology (NIST) Cybersecurity Risk Management Framework Applied to Modern Vehicles," Technical Report DOT HS 812 073, National Highway Traffic Safety Administration, Washington, DC, October 2014.

[46] V. L. Mihailescu, "Risk analysis and risk management using MEHARI," *J. Appl. Bus. Inf. Syst*, vol. 3, no. 4, pp. 143–161, 2012.

[47] A.-A. Affia, A. Nolte, and R. Matulevičius, "Security risk management in cooperative intelligent transportation systems: A systematic literature review," 10 2019.

[48] V. L. L. Thing and J. Wu, "Autonomous vehicle security: A taxonomy of attacks and defences," 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)*, pp. 164–170, 2016.

[49] Z. El-Rewini, K. Sadatsharan, D. F. Selvaraj, S. J. Plathottam, and P. Ranganathan, "Cybersecurity challenges in vehicular communications," *Vehicular Communications*, vol. 23, p. 100214, 2020.

[50] J. Petit and S. E. Shladover, "Potential cyberattacks on automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 546–556, 2015.

[51] M. N. Mejri, J. Ben-Othman, and M. Hamdi, "Survey on VANET security challenges and possible cryptographic solutions," *Vehicular Communications*, vol. 1, no. 2, pp. 53 – 66, 2014.

[52] S. Parkinson, P. Ward, K. Wilson, and J. Miller, "Cyber threats facing autonomous and connected vehicles: Future challenges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 11, pp. 2898–2915, 2017.

[53] T. Soiunen, "Design and evaluation of a user interface to increase trust in autonomous vehicles," Master's thesis, University of Tartu, Institute of Computer Science, Tartu, 2020.

[54] D. J. Bodeau, C. D. McCollum, and D. B. Fox, "Cyber threat modeling: Survey, assessment, and representative framework," technical paper, Homeland Security Systems Engineering and Development Institute (HSSEDI™), 2018.

[55] "The STRIDE Threat Model." https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20). Last accessed 23-Jun-2020.

[56] The MITRE Corporation, "CAPEC - Common Attack Pattern Enumeration and Classification." https://capec.mitre.org/. Last accessed 28-Feb-2021.

[57] The MITRE Corporation, "MITRE ATT&CK." https://attack.mitre.org/. Last accessed 28-Feb-2021.

[58] "OWASP Projects." https://owasp.org/projects/. Last accessed 4-Nov-2020.

[59] O. Altuhhov, R. Matulevičius, and N. Ahmed, "An extension of business process model and notation for security risk management," *Int. J. Inf. Syst. Model. Des.*, vol. 4, p. 93–113, Oct. 2013.

[60] K.-Y. Park, S.-G. Yoo, and J. Kim, "Security requirements prioritization based on threat modeling and valuation graph," in *Convergence and Hybrid Information Technology* (G. Lee, D. Howard, and D. Ślęzak, eds.), (Berlin, Heidelberg), pp. 142–152, Springer Berlin Heidelberg, 2011.

[61] D. Firesmith, "Engineering security requirements," *Journal of Object Technology*, vol. 2, pp. 53–68, January-February 2003.

[62] National Highway Traffic Safety Administration, "Cybersecurity Best Practices for Modern Vehicles (Report No. DOT HS 812 333)." https://www.nhtsa.gov/staticfiles/nvs/pdf/812333_CybersecurityForModernVehicles.pdf, October 2016. Last accessed 15-Oct-2020.

[63] "Critical security controls, version 7.1," tech. rep., Center for Internet Security, April 2019.

[64] S. International, "Sae j3061: Surface vehicle recommended prac-tice - cybersecurity guidebook for cyber-physical vehicle systems," tech. rep., 2016.

[65] J. T. Force, "Security and privacy controls for information systems and organizations," Tech. Rep. NIST Special Publication (SP) 800-53, Rev. 5, National Institute of Standards and Technology, Gaithersburg, MD, 2020.

[66] N. Mead, "Identifying security requirements using the security quality requirements engineering (square) method," in *Integrating Security and Software Engineering: Advances and Future Visions*, pp. 43–69, IGI Global, 2007.

[67] N. Ahmed and R. Matulevicius, "A method for eliciting security requirements from the business process models.," in *CAiSE (Forum/Doctoral Consortium)*, pp. 57–64, 2014.

[68] A. Pattakou, C. Kalloniatis, and S. Gritzalis, "Security and privacy requirements engineering methods for traditional and cloud-based systems: a review," *Cloud Comput*, vol. 2017, p. 155, 2017.

[69] "Privacy Policy for Passengers." https://bolt.eu/en/legal/privacy-for-riders/. Last accessed 24-Nov-2020.

[70] The MITRE Corporation, "Common Weakness Enumeration (CWE)." https://cwe.mitre.org/. Last accessed 28-Feb-2021.

[71] "A malicious app captures the raw screen buffer." https://pages.nist.gov/mobile-threat-catalogue/application-threats/APP-40.html#fn:AndroidDevBlog-1. Last accessed 22-Jul-2020.

[72] "An investigation of Chrysaor Malware on Android." https://android-developers.googleblog.com/2017/04/an-investigation-of-chrysaor-malware-on.html. Last accessed 22-Jul-2020.

[73] L. Stefanko, "Malware sidesteps Google permissions policy with new 2FA bypass technique." https://www.welivesecurity.com/2019/06/17/malware-google-permissions-2fa-bypass/. Last accessed 22-Jul-2020.

[74] A. Cetinkaya and H. Ishii, "Jamming attacks: A major threat to controlling over wireless channels." http://blog.ifac-control.org/technology/jamming-attack/jamming-attacks-a-major-threat-to-controlling-over-wireless-channels/. Last accessed 27-Jul-2020.

[75] J. H. Saltzer and M. D. Schroeder, "The protection of information in computer systems," *Proceedings of the IEEE*, vol. 63, no. 9, pp. 1278–1308, 1975.

[76] J. Serota and A. Irom, "Best practices for securing your API." https://www.imperva.com/blog/best-practices-for-securing-your-api/. Last accessed 15-Sep-2020.

[77] M. Tayyab Asghar, M. Riaz, J. Ahmad, and S. Safdar, "Security model for the protection of sensitive and confidential data on mobile devices," in *2008 International Symposium on Biometrics and Security Technologies*, pp. 1–5, 2008.

[78] T. Kiravuo, M. Sarela, and J. Manner, "A survey of Ethernet LAN security," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1477–1491, 2013.

[79] "OWASP cheat sheet series project, version 2.0," tech. rep., Open Web Application Security Project, 2019.

[80] K. Grover, A. Lim, and Q. Yang, "Jamming and anti–jamming techniques in wireless networks: a survey," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 17, no. 4, pp. 197–215, 2014.

[81] D. Mitropoulos and D. Spinellis, “Fatal injection: a survey of modern code injection attack countermeasures,” *PeerJ Computer Science 3*, November 2017.

[82] M. Martin and M. S. Lam, “Automatic generation of xss and sql injection attacks with goal-directed model checking,” in *Proceedings of the 17th Conference on Security Symposium*, SS’08, (USA), p. 31–43, USENIX Association, 2008.

Appendix

I. Glossary

API	Application Programming Interface
ATT&TK	Adversarial Tactics, Techniques, and Common Knowledge
AV	Autonomous Vehicle
BA	Business Asset
BPMN	Business Process Model Notation
CAPEC	Common Attack Pattern Enumeration and Classification
CIA	Confidentiality, Integrity, Availability
EU	European Union
FHE	Fully Homomorphic Encryption
GDPR	General Data Protection Regulation
HCI	Human-Computer Interaction
HRI	Human-Robot Interaction
IoT	Internet of Things
IS	Information System
ISSRM	Information Systems Security Risk Management
ISMS	Information Security Management System
ITS	Intelligent Transportation System
OWASP	Open Web Application Security Project
PET	Privacy-Enhancing Technology
PK	Public Key
RQ	Research Question
SRQ	Sub-Research Question
SDLC	System Development Lifecycle
SRM	Security Risk Management
STRIDE	Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, Elevation of privileges
UML	Unified Modelling Language
VANET	Vehicular Ad Hoc Network
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle

II. Ride Fulfilment Business Process Model

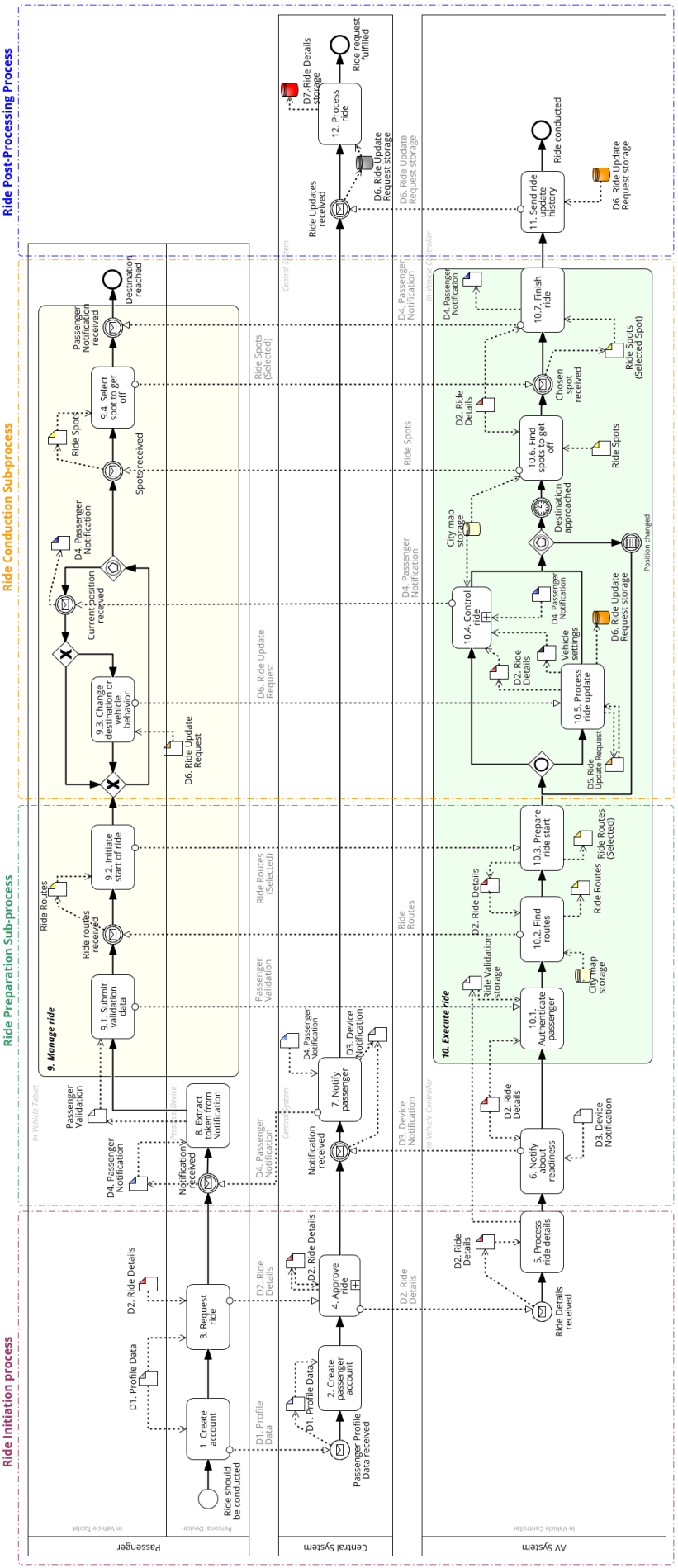

Figure 28. The detailed Ride Fulfilment business process model

III. Detailed Security Threat Model

Class	Threat	Characteristics	Impact
Spoofing	1. **ST1. Identity Spoofing:** An attacker who has explored the authentication procedure and received access to login user's interface, uses obtained credentials to authenticate himself as a legitimate user. - CAPEC-151 Identity Spoofing, 633: Token Impersonation, ATT&CK: T1078 Valid Accounts, T1098 Account Manipulation)	**V1:** Lack of full logging of authentication process (e.g. details of the machine used for authentication) that enables an attacker not to be revealed. **Threat Agent:** An attacker has knowledge about authorization procedure and revealed which user credentials can be used for the further attack. **Attack method:**1. An attacker has successfully obtained credentials of the targeted user. 2. An attacker gets access to login interface of passenger-to-system interaction and authenticates himself to the system as a legitimate user. 3. An attacker sends messaged which contains content that enables further attacks.	An adversary can access a system as a legitimate user, ergo, making the adversary harder to detect enables exploitation of others system vulnerabilities for further information disclosure, the elevation of privileges, or data altering. Also is can be a part of such attack as *Lateral Movement (TA0008).*
Tampering	2. **TT1. Data Storage Manipulation:** After the obtaining access to the system and authorised access to the targeted data storage, an attacker alters, discards or inserts data into the storage by exploiting lack of logging. - CAPEC-165: File Manipulation, ATT&CK: T1565.001 Stored Data Manipulation	**V1:** Poorly defined authentication permissions. (OWASP Top Ten: 5.*Broken Access Control*) **V2:** Lack of logging storage's data manipulation. (OWASP Top Ten: 10.*Insufficient Logging and Monitoring*) **V3:** The system use files from data storage without verifying integrity. **Threat Agent:** An attacker has knowledge about format of stored data and is can get authorized access to manipulate data storage. **Attack method:**1. An attacker has obtained access to the system. 2. An attacker obtained authorized access to manipulate data storage. 3. An attacker has alter, discarded or inserted data into the accessed data storage.	Compromises the integrity of the used data storage for ride conduction on AV that may affect business process and decision making. Another implementation of this attack is manipulating log files, that implies repudiation of the system.

III. Detailed Security Threat Model (Continued)

Class	Threat	Characteristics	Impact
Tampering	3. **TT2. API Parameters Manipulation:** As a result of successful Man-in-the-Middle attack, an attacker manipulates the use or handling used by the targeted system components API by tampering parameters. - CAPEC-113: API Manipulation	**V1:** Lack of integrity and validation of received parameters. **V2:** Usage of Test APIs on production systems as a part of active debug code [70]. **Threat Agent:** An attacker has knowledge about how the targeted system components use APIs and should be able to conduct reverse engineering of API syntax. **Attack method:** 1. An attacker implements Man-in-the-Middle attack in order to detect the manner in which the system uses an API. 2. An attacker may conduct reverse engineering the API syntax for defining possible attack vector. 3. An attacker manipulates the use or handling the API by manipulating parameters that are exchanged between client and server in order to modify application data.	Enables forcing a system component to send an unexpected request to API which is not validated and cannot be handled properly. Such manipulation can also lead to any kinds of injection flaws (see ET1).
	4. **TT3. Software Source Code Manipulation:** An attacker exploits neglecting of the downloaded code check by delivering malicious code to the system components as a part of authorized software update, ergo, negating integrity of the software. - CAPEC-184: Software Integrity Attack	**V1:** Neglecting integrity checking of the downloaded code. **V2:** Security bugs in firmware of the targeted system component. **Threat Agent:** An attacker has knowledge about the system infrastructure, components and deployment process (e.g. regular updates). An attacker may be required to have knowledge about how the system users can be manipulated. An attacker should be able to develop malicious code that the targeted system component will be able to run during its normal process flow. **Attack method:** 1. An attacker has successfully implemented early phase attack vectors and revealed the system infrastructure, components, processes of downloading source code (e.g. regular updates) and deployment. 2. An attacker has created code for malicious software update that negates integrity of the targeted system component. 3. An attacker introduces malicious code to the targeted system by means of modification the payload software update. 4. An attacker ensures running the uploaded malicious code (e.g. social engineering attack).	Implementation of malicious code may compromise integrity of software code, hardware system components, data structure or firmware enabling bringing the targeted asset into insecure state.

III. Detailed Security Threat Model (Continued)

Class	Threat	Characteristics	Impact
Repudiation	5. **RT1. System Log Files Manipulation:** An attacker with access to the data storage manipulated the system log files in order to hide another attack. - CAPEC-165: File Manipulation, ATT&CK: T1565.001 Stored Data Manipulation	**V1:** Poorly defined authentication permissions. (OWASP Top Ten: 5.*Broken Access Control*). **V2:** Lack of logging storage's data manipulation. (OWASP Top Ten: 10.*Insufficient Logging and Monitoring*) **Threat Agent:** An attacker with knowledge of the system architecture and processes within the system has intention to alter the targeted process flow. **Attack method:** 1. An attacker has obtained access to the system. 2. An attacker obtained authorized access to manipulate data storage. 3. An attacker has alter, discarded or inserted data into the log file.	Compromises integrity of the system log files that causes repudiation of the system components and helps an attacker to hide his other attacks to prevent being revealed.
Information Disclosure	6. **IT1. Device Screen Capturing:** An attacker with intention to gather data from the app on the targeted device exploits vulnerability of the device's firmware using the malicious app which after elevation of privileges tracks the screen of the mobile device. - Mobile ATT&CK T1513: Screen Capture	**V1:** The targeted person has weaknesses in his psychology so that it can be used as the target of social engineering attack. **V2:** Security bugs in the firmware of the device. **V3:** A mobile app to not apply FLAG_SECURE window flag for the app windows which are supposed to contain sensitive data. **Threat Agent:** An attacker has gathered information about the targeted person, so he identified people weaknesses and identified weaknesses of the targeted mobile app. **Attack method:** 1. An attacker uses social engineering attack to force an owner of the targeted device to install a malicious app. 2. A malicious app conducts rooting for elevation of privileges. 3. An attacker collects authentication credentials or get unauthorized access to sensitive data displayed in the foreground by means of the app which captures content of the screen buffers.	The described attack method was implemented in Android devices [71, 72], but the attack pattern is applicable for iOS devices as well. The threat enables disclosure of sensitive information that can be used for further spoofing attacks.

III. Detailed Security Threat Model (Continued)

Class	Threat	Characteristics	Impact
Information Disclosure	7. **IT2. Device Notification Manipulation:** An attacker with intention to gather data from the device notification exploits vulnerability installed apps using the malicious app which after elevation of privileges manipulate device notification in order to either gather sensitive data or reveal conduction of another attack. - Mobile ATT&CK T1517: Access Notifications	**V1:** The targeted person has weaknesses in his psychology so that it can be used as the target of social engineering attack. **V2:** The app includes sensitive information into the device notification. **Threat Agent:** An attacker has gathered information about the targeted person that allowed him to identify person's weaknesses; an attacker identified weaknesses of the targeted mobile app. **Attack method:** 1. An attacker uses a social engineering attack to force an owner of the targeted device to install a malicious app (Mobile ATT&CK T1475, T1476). 2. The malware requests a user for *Notification access* permissions. 3. The malware is running on background and either captures incoming device notifications in order to collect sensitive data or conduct another attack dismissing incoming notifications.	Helps to obtain sensitive information for the immediate usage or collection of data for its further analysis. The malicious app can not only access the content of the targeted apps on the victim's device, but also dismiss incoming notification that enables effective bypassing two-factor authentication (2FA) [73].
	8. **IT3. Access Stored Application Data:** An attacker with intention to get access to sensitive data, produced by the mobile app, revealed that such data is stored locally on a device, ergo, he accesses the data using authorized access or exploits weakness of storing data in an insecure manner. - Mobile ATT&CK T1409: Access Stored Application Data	**V1:** The app stores sensitive data in an insecure manner or with broken authorization, ergo, an attacker can access it without elevation of privileges (OWASP Top Ten: 3. *Sensitive data exposure*). **Threat Agent:** An attacker with has access to the device file system and has knowledge about the structure of app created files **Attack method:** 1. An attacker has obtained access to the device file system. 2. An attacker exploits lack of storing data in an insecure manner and read the content of the files.	The attack may leads to revealing such sensitive information such as user's credentials or history of user activities within the application.

III. Detailed Security Threat Model (Continued)

Class	Threat	Characteristics	Impact
Information Disclosure	9. **IT4. Man-in-the-Middle Attack:**An attacker implement Man-in-the-Middle attack placing himself in the communication channel between the two system components in order to passively or actively listen to the data flows. - Man-in-the-Middle (ATT&CK T1557 or CAPEC-94)	**V1:**The transmission channel is not protected with mutual authentication. (OWASP Top Ten: 2.*Broken Authentication*) **V2:** The communication between system components is conducted without data encryption. (OWASP Top Ten: 3. *Sensitive Data Exposure*) **Threat Agent:** An attacker is able to explore nature and mechanism of communication between the system components and has access to the targeted transmission channel. **Attack method:** 1. An attacker determines mechanism of communication between the system components. 2. An attacker posses himself into the communication channel as a proxy between the targeted system components. 3. An attacker observes data transmitted between the system components with or without its altering for the further its usage.	Supports *Network Sniffing (CAPEC-158, T1040)* or *Transmitted Data Manipulation (T1565.002)* attacks
Denial of Service	10. **DT1. Communication Channel Jamming:** An attacker with ability to use signal emitter jams the targeted communication channel (Wi-Fi or cellular) using radio noise or signals with intention to prevent communication. - CAPEC-604: Wi-Fi Jamming, 605 Cellular Jamming, Mobile ATT&CK T1464: Jamming or Denial of Service	**V1:** Lack of anti-jam functionality in Wi-Fi and cellular technologies. (OWASP: 9. *Using Components with Known Vulnerabilities*) **Threat Agent:** An attacker does not required to have knowledge about internals of the control system [74]. An attacker has revealed that the system use the technology which experiences the lack of anti-jam features and/or authentication on deauthentication packets in case of Wi-Fi jamming [56]. An attacker should have signal emitter. **Attack method:** 1. An attacker has successfully discovered the technology used in communication channel. 2. An attacker produces signals (e.g. noise on the radio frequency band used by the Wi-Fi network) in order to prevent users or system components from communicating within the targeted network.	May lead to partial or full full blockage of the normal system work as a consequence of effective blocking communication of the system components. As a result, a business process can be either interrupted and stopped or cause some kind of damage depending on how errors handling is implemented within the system.

III. Detailed Security Threat Model (Continued)

Class	Threat	Characteristics	Impact
Denial of Service	11. **DT2. Forced Service Deadlock:** An attacker with knowledge of the system resources and used APIs initiates running of two or more competing actions which cannot finish due to the improper synchronization that results in a deadlock state of the system. - CAPEC-25: Forced Deadlock	**V1:** Improper synchronization of processes and a lock on resources[70]. **V2:** Unsynchronized access to shared data in a multithreaded context [70] **Threat Agent:** An attacker has knowledge about the system resources and used APIs as well as expertise in revealing deadlock condition. **Attack method:** 1. An attacker conducts investigation phase to gain the knowledge about the targeted system. 2. An attacker initiated the start of the first action to hold the resource and triggers the second action which is waiting for the held resource. 3. If the system does not have the proper handling of the synchronized processes operation which leads to a deadlock condition, the targeted system runs the processes continuously which cases a denial of service.	May negated availability of the system components or data which can be the reason of the failure of the normal business process flow or leads to the full denial of the system operating.
	12. **DT3. Service Flooding:** An attacker exploits improper resource allocation and resources release by consuming the resources of a target as a result of tremendous amount of interactions with the target that prevents the ability of a target to communicated with other users. - CAPEC-125: Flooding, ATT&TK T1499: Endpoint Denial of Service	**V1:** Lack of limitation during resources allocation (CWE-770). **V2:** Improper resource shutdown or release (CWE-404). **Threat Agent:** An attacker is able to run script or program for generating more requests than the targeted service is able to handle. Alternatively, an attacker may use a network of machines that can produce requests concurrently. An attacker may be required to conduct session spoofing in advance to be legitimate to send requests. **Attack method:** 1. An attacker has gained knowledge about existing vulnerabilities of the used protocols and resources allocation. 2. An attacker acquires authorized access to send requests to the targeted service. 3. An attacker triggers flow of requests to the service which is not able to handle them due to the high amount of required resources.	The attack is focused on consuming resources on handling tremendous amount of received by the service requests and inability to handle other requests from legitimate users. Depending on the used communication channels, an attack can be implemented by means of corresponding protocol requests (e.g. TCP, HTTP, XML, etc.).

III. Detailed Security Threat Model (Continued)

Class	Threat	Characteristics	Impact
Denial of Service	13. **DT4. HTTP Flooding:** An attacker with knowledge about existing vulnerabilities of the used HTTP protocol and resources allocation, conducts flooding attack at the HTTP level so that resources are held and the HTTP session is kept alive during the essential time waiting for the response from the request sender preventing the legitimate user from setting connection with the service. - CAPEC-469: HTTP DoS , ATT&CK T1499: Endpoint Denial of Service	**V1:** Lack of limitation during resources allocation (CWE-770). **V2:** Lack of effective resource releasing after the end of effective lifetime (CWE-772). **Threat Agent:** An attacker is able to conduct flooding attack at the HTTP level and is able to produce requests which will hold the resources and keep the HTTP session alive during essential time. **Attack method:** 1. An attacker has gained knowledge about existing vulnerabilities of the used HTTP protocol and resources allocation. 2.An attacker triggers the flow of initial requests (analog to SYN messages) to the service so that resources are held and the HTTP session is kept alive during the essential time waiting for the response from the request sender preventing the legitimate user from setting connection with the service.	In contrast to the HTTP flooding attack (see ET3), the attack does not depend upon a large number of requests, but relies on the repeated requests that keep the resources help, that leads to inability of legitimate users get access to the service.

III. Detailed Security Threat Model (Continued)

Class	Threat	Characteristics	Impact
Elevation of Privileges	14. **ET1. Command Injection:** After successful system analysis, an attacker provides a malicious command as input for the system which leads to XML, SQL or other kind of injections in order to make it possible to reach the targeted asset for further attacks implementation. - CAPEC-248: Command Inject, ATT&CK T1220: XSL Script Processing, T1190: Exploit Public-Facing Application	**V1:** Any kind of insufficient implementations of the web interfaces (e.g. validation of supplied by user data). **V2:** Insecure command construction. **Threat Agent:** An attacker has knowledge about the targeted network structure, its environment, vulnerabilities of input validation. **Attack method:** 1. An attacker has successfully conducted the system analysis and defined the system structure. 2. An attacker provides a malicious command as an input for the system which leads to XML, SQL or other kind of injections. 3. An attacker obtain access to the targeted asset.	Enables execution of adversary's command which acts as an intermediate phase for getting obtaining access and enabling further attacks (e.g. *Stored/Runtime Data Manipulation (TT1565.001/003), CAPEC-255: Manipulate Data Structures*).
	15. **ET2. Bypass API Authentication:** An attacker with intention to reach the system functionality or data has found an endpoint which enables access to the targeted asset evading or avoiding authentication. - CAPEC-115: Authentication Bypass, CAPEC-121: Exploit Test APIs	**V1:** Improper authentication mechanism within the system enables access to the specific functionality (access to the task within the process bypassing the previous steps) or data without checking an identity (OWASP Top Ten: *5. Broken Access Control*). **V2:** Usage of Test APIs on production systems as a part of active debug code [70]. **Threat Agent:** An attacker has experience in API structure detection and has intention to reach the functionality of the system by evading or avoiding an authentication mechanism. **Attack method:** 1. An attacker had researched the system component and used APIs and discovered the way to reach the targeted asset bypassing authentication. 2. An attacker exploits discovered weakness by reaching the targeted endpoint of the system without any authentication.	Enables access to sensitive data or crucial functionality (e.g. access to sensitive information or administrative intended functionality) that helps to conduct further attacks of information disclosure or system repudiation.

III. Detailed Security Threat Model (Continued)

Class	Threat	Characteristics	Impact
Elevation of Privileges	16. **ET3. Session Hijacking:** An attacker has explored the authentication procedure and hijacked the session in order to spoof passenger's identity. - CAPEC-21: Exploitation of Trusted Credentials, ATT&CK T1539: Sblack Web Session Cookie	**V1:** Improper sessions management of a web-based application component for interaction with a vehicle passenger. (OWASP Top Ten: 2.*Broken Authentication*) **Threat Agent:** An attacker has knowledge about session management within the targeted system and has an ability to conduct session hijacking. **Attack method:** 1. An attacker has successfully conducted network sniffing or XSS attack and, as a result, has obtained a valid token session generated during the vehicle passenger authentication. 2. An attacker uses stolen session ID to imitate a passenger's behaviour and control the AV.	An adversary can access a system as a legitimate user, ergo, making the adversary harder to detect enables exploitation of others system vulnerabilities for further information disclosure, the elevation of privileges, or data altering. Also is can be a part of such attacks as *Identity Spoofing (CAPEC-151), Lateral Movement (TA0008).*
	17. **ET4. Credentials Brute Force:** An attacker who has explored the authentication procedure and received access to login user's interface conducts credentials brute forcing in order to spoof passenger's identity. - *Brute Force* (CAPEC-112, ATT&CK: T1110)	**V1:** Authentication mechanism allows to conduct authentication without limited number of attempts **V2:** Passenger authentication can be executed from more than one device within the LAN. **Threat Agent:** An attacker has knowledge about a general structure of credentials used for passenger authentication and is able to execute brute force attack. **Attack method:**1. Obtain knowledge about method and used credential structure for user authentication by determining secret testing procedure, reducing search space, gathering information if an attack can be performed independently. 2. Get access to login interface of passenger-to-system interaction. 3. Implement credential brute force attack.	An adversary can access a system as a legitimate user, ergo, making the adversary harder to detect enables exploitation of others system vulnerabilities for further information disclosure, the elevation of privileges, or data altering. Also is can be a part of such attacks as *Identity Spoofing (CAPEC-151), Lateral Movement (TA0008).*

IV. Security Risks

	SR1. Passenger Identity Spoofing
Business Asset	Passenger Notification, Ride Details, Ride Update Request
System Asset	**Passenger** entity
Vulnerability	Lack of full logging of authentication process (e.g. details of the machine used for authentication) in **In-Vehicle Controller**. **Passenger** authentication in **Web Client** can be executed from more than one device within the LAN.
Threat	ST1. An attacker who has explored the authentication procedure and received access to login user's interface, uses obtained credentials to authenticate himself as a legitimate Passenger.
Impact	Compromises **confidentiality** of **Passenger Notification** and as a consequence **confidentiality** of **Ride Details** as Passenger Notification contains Current Position and other details; compromises reliability (i.e. **integrity**) of any **Ride Update Request**)
Risk	An attacker who has explored the authentication procedure and received access to login user's interface, uses obtained credentials to authenticate himself as a legitimate Passenger that compromises confidentiality of Passenger Notification, and as a consequence Ride Details as well as loose of reliability (i.e. integrity) of any Ride Update Request.

IV. Detailed Security Risk Model (Continued)

	TR1. City Map Storage Manipulation	**TR2. Ride Validation Storage Manipulation**	**TR3. Ride Update Request Storage Manipulation**
Business Asset	Ride Routes, Ride Spots	Passenger Validation	Ride Update Request (includes Ride Details Update and Vehicle Behavior Update)
System Asset	In-Vehicle Controller, City Map Storage	In-Vehicle Controller, Ride Validation Storage	In-Vehicle Controller, Ride Update Request storage
Vulnerability	Exploits lack of logging in **In-Vehicle Controller** and usage of files from **City map storage** without verifying its integrity	Exploits lack of logging in **In-Vehicle Controller** and usage of files from **Ride Validation storage** without verifying integrity.	Exploits lack of logging in **In-Vehicle Controller** and usage of files from **Ride Update Request storage** without verifying its integrity.
Threat	TT1. After the obtaining access to City Map storage, an attacker alters, discards or inserts data into the storage by exploiting lack of logging and and usage of files from City map storage without verifying its integrity.	TT1. After the obtaining access to the system and authorised access to Ride Validation storage, an attacker alters, discards or inserts data into the storage by exploiting lack of logging and usage of files from Ride Validation storage without verifying its integrity.	TT1. After the obtaining access to the system and authorised access to Ride Update Request storage, an attacker alters, discards or inserts data into the storage by exploiting lack of logging and usage of files from Ride Update Request storage without verifying its integrity.
Impact	Compromises **integrity** of **Ride Routes** (e.g. a set of roads were deleted from the map), **Ride Spots**	Compromises **integrity** of **Passenger Validation**	Compromises **integrity** of **Ride Update Request** represented by Ride Details Update and Vehicle Behavior Update
Risk	After the obtaining access to the system and authorized access to City map storage, an attacker alters, discards or inserts data into the storage by exploiting lack of logging and usage of files from storage without verifying its integrity leading to loss of availability and integrity of Ride Routes and Ride Spots.	After the obtaining access to the system and authorized access to Ride Validation storage, an attacker discards or alters data into the storage by exploiting lack of logging and usage of files from storage without verifying its integrity leading to loss of integrity of Passenger Validation.	After obtaining access to the system and authorized access to Ride Details Update storage, an attacker discards or alters data into the storage by exploiting the lack of logging and usage of files from storage without verifying its integrity that leads to loss of integrity of Ride Update Request (represented by Ride Details Update and Vehicle Behavior Update).

IV. Detailed Security Risk Model (Continued)

	TR4. Central IS API Parameters Manipulation	**TR5. User Device Controller Source Code Manipulation**	**TR6. In-Vehicle Controller Source Code Manipulation**
Business Asset	Device Notification, Passenger Notification	Device Notification, Passenger Notification	Ride Details,Vehicle Settings, Ride Update Request, Ride Routes,Passenger Notifications, Ride Spots
System Asset	Central IS API, User Device Controller	User Device Controller	In-Vehicle Controller
Vulnerability	Lack of integrity and validation of received parameters by **User Device Controller** and **Central IS API**; usage of **Central IS Test API** on production systems as a part of active debug code	Lack of integrity checking of the downloaded code by **User Device Controller** and/or existence of security bugs in firmware of the Bolt system components.	Lack of integrity checking of the downloaded code by **In-Vehicle Controller** and/or existence of security bugs in firmware of the Bolt system components.
Threat	TT2. As a result of a successful Man-in-the-Middle attack, an attacker manipulates the use or handling of Central IS API by tampering exchanged parameters.	TT3. An attacker exploits neglecting of the downloaded code check by delivering malicious code to User Device Controller as a part of an authorized software update, ergo, negating integrity of the software.	TT3. An attacker exploits the neglect of the downloaded code check by delivering malicious code to the system components as part of an authorized software update, ergo, negating the software's integrity.
Impact	Compromises **integrity** of **Device Notification** that can leads to loose of **integrity** of **Passenger Notification** as it uses Device Notification as a data source	Compromises **integrity** and availability of **Passenger Notifications** and **Device Notification** as well as **confidentiality** of **Passenger Notifications**	Compromises **confidentiality**, **integrity** and **availability** of **Ride Details** as well as **integrity** and **availability** of **Ride Update Request**, **Ride Routes**, **Vehicle Settings**, **Passenger Notifications** and **Ride Spots**
Risk	As a result of a successful Man-in-the-Middle attack, an attacker manipulates transferred to Central IS API parameters, which are not properly validated by User Device Controller that compromises the integrity of Device Notification, and as a consequence - the integrity of Passenger Notification as it is extracted from the tampered Device Notification.	An attacker exploits neglecting of the downloaded code check by delivering malicious code to User-Device Controller as a part of an authorized software update, ergo, negating integrity of the software which enables manipulating any business assets User-Device Controller has access to.	An attacker exploits neglecting of the downloaded code check by delivering malicious code to In-Vehicle Controller as a part of an authorized software update, ergo, negating integrity of the software which enables manipulating any business assets In-Vehicle Controller has access to.

IV. Detailed Security Risk Model (Continued)

	RR1: System Log Files Manipulation	**IR1. Personal Mobile Device Screen Capturing**	**IR2. In-Vehicle Tablet Device Screen Capturing**
Business Asset	Ride Details	Passenger Notification	Passenger Notification
System Asset	In-Vehicle Controller	Mobile Client, Personal Mobile Device	Web Client, In-Vehicle Tablet
Vulnerability	Poorly defined authentication permissions in **In-Vehicle Controller**; lack of logging storage's data manipulation within **In-Vehicle Controller**	Security bugs in the firmware of the device and insecure configuration of **Mobile Client**	Security bugs in the firmware of the In-Vehicle Tablet and insecure displaying of confidential data on **Web Client**
Threat	RT1. An attacker with access to the System Log Files manipulates them in order to hide another attack during which Ride Details were altered.	IT1. An attacker with intention to gather data from the app on the targeted device exploits vulnerability of the device's firmware using the malicious app which after elevation of privileges tracks the screen of the mobile device.	IT1. An attacker with intention to gather data from the web browser app on the In-Vehicle Tablet exploits vulnerability of the tablet firmware using the malicious app which after elevation of privileges tracks the screen.
Impact	Compromises **integrity** of **Ride Details**	Compromises **confidentiality** of **Passenger Notification**	Compromises **confidentiality** of **Passenger Notification**
Risk	An attacker with access to the System Log Files in In-Vehicle Controller manipulates them to hide the change of Ride Details internally by malicious code of In-Vehicle Controller.	An attacker with intention to obtain credentials from the app on the targeted device exploits vulnerability of the personal mobile device's firmware and insecure configuration of Mobile Client by means of the malicious screen capturing app that negates confidentiality of Passenger Notification.	An attacker with intention to obtain credentials from the web browser on the in-vehicle tablet exploits vulnerability of the device's firmware and displaying of confidential data on Web Client by means of the malicious screen capturing app that negates confidentiality of Passenger Notification.

IV. Detailed Security Risk Model (Continued)

	IR3. Personal Mobile Device Notification Manipulation	**IR4. Access Stored Application Data**	**IR5. Man-in-the-Middle**
Business Asset	Passenger Notification	Passenger Notification	Passenger Notification
System Asset	Mobile Client	Mobile Client	Transmission channel used for Web-Client to In-Vehicle Controller communication
Vulnerability	**Mobile Client** includes sensitive information into the device notification which contains data from Passenger Notification	**Mobile Client** which stores sensitive data in an insecure manner or with broken authorization	The **transmission channel** used for Web-Client to In-Vehicle Controller communication is not protected with mutual authentication; The communication between system components is conducted without data encryption.
Threat	IT2. An attacker with intention to gather data from the device notification exploits vulnerability installed apps using the malicious app which after elevation of privileges manipulate device notification in order to either gather sensitive data or reveal conduction of another attack.	IT3. An attacker with intention to get access to sensitive data, produced by the mobile app, revealed that such data is stored locally on a device, ergo, he accesses the data using authorized access or exploits weakness of storing data in an insecure manner.	IT4. An attacker places himself in the communication channel between Web-Client and In-Vehicle Controller to passively or actively listen to the transferred data flows.
Impact	Compromises **confidentiality** and **availability** of **Passenger Notification**	Compromises **confidentiality** of **Passenger Notification**	Compromises **confidentiality** of **Passenger Notification**
Risk	An attacker with intention to manipulate device notifications on the Personal Mobile Device exploits vulnerability of the Mobile Client to include sensitive data in device notification by means of the malicious app that negates confidentiality of Passenger Notification.	An attacker with intention to get access to Passenger Notification from the Mobile Client app, revealed that such data is stored locally on a device, ergo, he accesses the data using authorized access or exploits weakness of storing data in an insecure manner, so an attacker is able to identify route which was used for the ride.	An attacker places himself in the transmission channel between Web-Client to In-Vehicle Controller to passively listen to the transferred data flows and exploit the lack of data encryption that leads to compromising the confidentiality of Passenger Notification.

IV. Detailed Security Risk Model (Continued)

	DR1. Front-Back Communication Channel Jamming	**DR2. Back-Ends Communication Channel Jamming**	**DR3. Forced In-Vehicle Controller Deadlock**
Business Asset	Passenger Notification, Passenger Validation, Ride Routes, Ride Update Request	Device Notification	Ride Routes, Ride Spots
System Asset	Transmission channel used for Passenger UI Client to back-end components (User-Device Controller/In-Vehicle Controller) communication	Transmission channel used for communication between In-Vehicle Controller and User-Device Controller	City Map Storage, In-Vehicle Controller
Vulnerability	Targets sensitivity of used **Transmission channels** to unexpected radio noise or signals and lack of anti-jam functionality in Wi-Fi and cellular technologies	Targets sensitivity of used **Transmission channels** to unexpected radio noise or signals and lack of anti-jam functionality in Wi-Fi and cellular technologies	Improper synchronization of processes which use **City Map storage** and a lack on resources by **In-Vehicle Controller**
Threat	DT1. An attacker with ability to use signal emitter jams the targeted communication channel (Wi-Fi or cellular) using radio noise or signals with intention to prevent communication.	DT1. An attacker with ability to use signal emitter jams the targeted communication channel (Wi-Fi or cellular) using radio noise or signals with intention to prevent communication.	DT2. An attacker revealed that In-Vehicle Controller cannot use City Map Storage (lack of a local version of required for the ride conduction the map) while it is updates by Central IS, so he forced update of the City Map Storage by implementing another attack on Central IS)
Impact	Compromises **availability** of any data transferred through the **Transmission channels**: **Passenger Notification**, **Passenger Validation**, **Ride Routes**, **Ride Update Request** (represented by **Ride Details Update** and **Vehicle Behavior Update**), **Ride Spots**	Compromises **availability** of **Device Notification**	Compromises **availability** of **Ride Routes** or **Ride Spots**
Risk	An attacker with ability to use signal emitter jams the targeted communication channel used for communication between User-Device Controller/In-Vehicle Controller and Passenger UI Client using radio noise or signals with intention to prevent their communication.	An attacker with ability to use signal emitter jams the targeted communication channel used for communication between In-Vehicle Controller and User-Device Controller using radio noise or signals with intention to prevent their communication	An attacker revealed that In-Vehicle Controller cannot build Ride Routes/Ride Spots while City Map Storage is updates by Central IS, so an attacker forced update of the storage by implementing another attack, that leads to missing availability of Ride Routes/Ride Spots and the whole process termination.

IV. Detailed Security Risk Model (Continued)

	DR4. Service Flooding	**DR5. HTTP Flooding**	**ER1. Command Injection**
Business Asset	Ride Update Request	Passenger Validation	Passenger Validation
System Asset	In-Vehicle Controller	Transmission Channel used for communication between Web Client and In-Vehicle Controller	AV System API, In-Vehicle Controller
Vulnerability	Lack of limitation during resources allocation; Improper resource shutdown or release within **In-Vehicle Controller**	Lack of effective resource releasing after the end of effective lifetime within **transmission channel**	Unsafe realisation of **AV System API**, insecure command construction and/or lack of validation of supplied by user data within **In-Vehicle Controller**
Threat	DT3. An attacker exploits improper resource allocation and resources release by consuming the resources of In-Vehicle Controller as a result of tremendous amount of interactions with it, that prevents the ability of In-Vehicle Controller to communicated with other users	DT4. An attacker with knowledge about existing vulnerabilities of the used HTTP protocol and resources allocation, conducts flooding attack at the HTTP level so that resources are held and the HTTP session is kept alive during the essential time waiting for the response from the request sender preventing the legitimate user from setting connection with the service for conducting validation.	ET1. After successful system analysis, an attacker provides a malicious command as input for the system which leads to XML, SQL or other kind of injections in order to make it possible to reach the targeted asset for further attacks implementation.
Impact	Compromises **availability** of **Ride Update Request**	Compromises **availability** of **Passenger Validation**	Compromises **integrity** of **Passenger Validation** with intention to compromise security criteria of other assets that use Passenger Validation as a data source
Risk	An attacker exploits improper resource allocation and resources release by consuming the resources of In-Vehicle Controller as a result of tremendous amount of fake sent Vehicle Behavior Update requests, that prevents the ability of In-Vehicle Controller to receive real Ride Update Requests from the legitimate Passenger	An attacker with knowledge about existing vulnerabilities of the used HTTP protocol and resources allocation during the passenger validation activity, conducts flooding attack at the HTTP level so that resources are held and the HTTP session is kept alive during the essential time waiting for the response from the request sender preventing the legitimate Passenger from setting connection with the service for conducting validation.	After successful system analysis, an attacker provides a malicious command as Passenger Validation input for AV System API and as a resource for In-Vehicle Controller which leads to XML, SQL or other kind of injections in order to make it possible to reach the main targeted asset.

IV. Detailed Security Risk Model (Continued)

	ER2. Bypass API Authentication	**ER3. Passenger Session Hijacking**	**ER4. Passenger Credentials Brute Force**
Business Asset	Ride Update Request	Ride Details Update	Passenger Notification
System Asset	In-Vehicle Controller, AV System API	**Passenger** entity; **Web Client** of AV System	**Web Client** of AV System
Vulnerability	Improper authentication mechanism within **In-Vehicle Controller** enables access to the specific functionality without checking an identity; usage of Test version of**AV System API** on production systems as a part of active debug code	Improper sessions management of a **Web Client** of AV System	Authentication mechanism in **Web Client** of AV System allows to conduct authentication without limited number of attempts
Threat	ET2. An attacker with intention to reach the system functionality or data has found an endpoint which enables access to the targeted asset evading or avoiding authentication.	ET3. An attacker has explored the authentication procedure within a Web Client and hijacked the session to spoof the passenger's identity and conduct activity from his account.	ET4. An attacker who has explored the authentication procedure and received access to login user's interface in Web Client conducts credentials brute forcing in order to spoof passenger's identity by defining Token Code from Passenger Notification.
Impact	Compromises **integrity** of **Ride Update Request** (represented by Ride Details update and Vehicle behavior update)	Compromises **integrity** of **Ride Details Update**	Compromises **confidentiality** of **Passenger Notification**
Risk	An attacker with the intention to change the behavior of the AV exploits an endpoint in AV System API by sending requests with Ride Update Request evading or avoiding authentication that negates integrity of Ride Details update / Vehicle behavior update as it loses reliability of received requests.	Having the session ID of the targeted Passenger, an attacker hijacks the session in a Web Client of AV System and spoofs the Passenger's identity, after which he compromises the integrity of Ride Details Update by sending the malicious one with the desired ride destination.	An attacker who has explored the authentication procedure conducts credentials brute-forcing in Web Client to spoof the Passenger's identity by defining Token Code from Passenger Notification that compromises the confidentiality of Passenger Notification exploiting authentication mechanism in Web Client.

V. Security Requirements and Controls

Note: P - preventive, D - detective, C - corrective; the term "*system*" refers to the system(s) used in AV for raid-hailing Fulfilment

Security Requirement	Class	Security Control Components
ST1. Identity spoofing.		
ST1.R1. The system should authenticate user.	P	**ST1.C1.P.** Multi-factor authentication for users [63, 57, 40].
		ST1.C2.P. Biometric authentication for users [48].
ST1.R2. The system should store user credentials securely.	P	**ST1.C3.P.** Credentials management policy (e.g., application development guidance; password storage rules [57], etc.).
ST1.R3. The system should separate different user roles.	C	**ST1.C4.C.** Roles separation policy [57, 63].
ST1.R4. The system should implement access control mechanism.	P	**ST1.C5.P.**Principle of least privilege to system functions for user roles; combination of authentication and authorization procedures for access control [57, 75].
ST1.R5. The system should log access attempts to its interfaces.	D	**ST1.C6.D.** Web API access logging [76].
ST1.R6. The organization should follow user management policy.	P	**ST1.C7.P.** Password management procedures (e.g., change password when the organization staff leave, or if there is believe that they have been compromised [58], etc.).
ST1.R7. The system should use secure protocols for provisioning of credentials.	P	**ST1.C8.P.** MitM-resistant protocols: SSL/TLS protocol [63], IPSec protocol suite [65].
TT1. Data Storage Manipulation.		
TT1.R1. The system should follow data management policy.	C	**TT1.C1.C.** Data back up procedure (e.g., automatic back up schedule [63]).
TT1.R2. The system should authenticate data storage users.	P	**TT1.C2.P.** Database authentication procedure (including authentication for connections from the local server [58]).
TT1.R3. The system should communicate with data storage via channel protected by encryption protocol.	P	**TT1.C3.P.** SSL/TLS or SSH protocol for accessing data storage [58].
TT1.R4. The system should authorize access for data storage users.	P	**TT1.C4.P.** Data storage access control (e.g., give access to the storage only from the backend application server [58]).

V. Security Requirements (Continued)

Security Requirement	Class	Security Control Components
TT2. API Parameters Manipulation.		
TT2.R1. The system interfaces should hide sensitive data during external communications.	P	**TT2.C1.P.** Encryption of sensitive data in the API's outputs [76].
TT2.R2. The system components should follow quality policy.	P	**TT2.C2.P.** API quality management policy [56].
TT2.R3. The system should validate input data.	C	**TT2.C3.C.** Data input validation based on its properties (e.g., length, type of input, acceptable values, missing or extra inputs, syntax, consistency across related fields, and conformance to business rules [58, 70]).
		TT2.C4.C. Server-side input validation [58].
		TT2.C5.C. Syntactically and semantically validation of input data (in that order) before using it [58].
TT3. Software Source Code Manipulation.		
TT3.R1 The system should remain integral after software updates.	P	**TT3.C1.P.** Software updates validation [56].
TT2.R2. The system should execute only authorized programs.	P	**TT3.C2.P.** Application control for regulating external files execution [57].
		TT3.C3.P. Application whitelisting technology [63].
TT2.R3. The system should follow policy of external systems quality.	P	**TT3.C4.P.** Policy of external software quality (e.g., use only system currently supported and receiving vendor updates [63]).
TT2.R4. The system should use a channel protected by encryption protocol for remote managing.	P	**TT2.C5.P.** SSH protocol for remote system control [58].
RT1. System Log Files Manipulation.		
RT1.R1. The system should use backups for recovering organizational information.	P	**RT1.C1.P.** Regular system data backups creation which are stored outside the system [57, 63].
RT1.R2. The organisation should ensure data backups protection.	P	**RT1.C2.P.** Physical security and encryption of data backups [63].
RT1.R3. The system should conduct action logging on the system components.	D	**RT1.C3.D.** Local logging on all systems and network devices [57].
RT1.R4. The system should maintain timestamps consistency of logs on system components.	P	**RT1.C4.P.** Usage of three synchronized time sources from which all servers and network devices retrieve time information on a regular basis [63].
RT1.R5. The system shold have central log management control.	D	**RT1.C5.D.** Aggregating logs for analysis and review [63].

V. Security Requirements (Continued)

Security Requirement	Class	Security Control Components
RT1.R6. The security personnel should analyse system logs according to the log management policy.	D	**RT1.C6. D.** Identification of anomalies or abnormal events [63].
IT1. Device Screen Capturing.		
IT1.R1. The system should protect displayed on the mobile device sensitive data.	P	**IT1.C1.P.** FLAG_SECURE on sensitive screens within an Android app [57].
IT1.R2. The system shall guide users to set particular configuration settings on the mobile devices used for interaction with the system.	P	**IT1.C2.P.** Advise not to grant consent for screen capturing [57].
IT2. Device Notification Manipulation.		
IT2.R1. The system should protect sensitive data provided to the mobile app.	P	**IT2.C1.P.** Include only non-sensitive data in the app notification text [57].
IT2.R2. The system shall guide users to set particular configuration settings on the mobile devices used for interaction with the system.	P	**IT2.C2.P.** Advise not to grant consent for device manipulation [57].
IT3. Access Stored Application Data.		
IT3.R1. The system should store only required for normal functioning sensitive data.	P	**IT3.C1.P.** Remove sensitive data not regularly accessed by organization from the network [63].
IT3.R2. The system should encrypt sensitive data stored on mobile devices.	P	**IT3.C2.P.** Approved by authorities cryptographic mechanisms for data storage on mobile devices [63] (e.g., AES algorithm for mobile file system manager [77]).
IT3.R3. The system should authenticate and authorize access to the data stored on mobile device.	P	**IT3.C3.P.** Secondary authentication mechanism not integrated in the system [63]
IT3.R4. The system should follow the cache policy for the web applications.	P	**IT3.C4.P.** Restrictive cache directives for the web traffic exchanged through HTTP and HTTPS [58] (e.g., the *Cache-Control*, *Pragma* HTTP headers or equivalent META tags [58]).
IT4. Man-in-the-Middle Attack.		
IT4.R1. The system should verify integrity of the transmitted data.	D	**IT4.C1.D.** Cryptographic hash functions: message authentication code (MAC) algorithms [48, 65], digital signature, and checksums [65].
		IT4.C2.D. Integrity verification mechanism for traffic only within the Ethernet LAN [78].
IT4.R2. The system should identify unauthorised connections to the network.	D	**IT4.C3.D.** Authenticated application layer proxy for the network traffic that comes goes to or from the Internet [63].

V. Security Requirements (Continued)

Security Requirement	Class	Security Control Components
IT4.R3. The system should ensure the confidentiality of transmitted information.	P	**IT4.C4.P.** Cryptographic mechanisms: SSL/TLS protocol [63], IPSec protocol suite [65].
		IT4.C5.P. Advanced encryption standard (AES) to encrypt wireless data in transit [63].
IT4.R4. The organization should control physical access to transmission channel within organizational facilities.	P	**IT4.C6.P.** Wiretapping sensor [65].
		IT4.C7.P. Locked wiring closet [65].
		IT4.C8.P. Protection of cabling by conduit or cable trays [65].
IT4.R5. The system should authenticate device before establishing connection.	P	**IT4.C9.P.** Bidirectional cryptographically based authentication [65].
IT4.R6. The system should follow the wireless capabilities policies.	P	**IT4.C10.P.** Usage of wireless networking capabilities only for essential functions [63, 65].
IT4.R7. The system should protect the entire web sessions.	P	**IT4.C11.P.** Encrypted HTTPS (TSL) protocols [79].
DT1. Communication Channel Jamming.		
DT1.R1. The system should have backup channel for communication.	C	-
DT1.R2. The system should use anti-jamming techniques.	P	**DT1.C2.C.** Spatial retreats approach[80] in the wireless network.
DT2. Forced Service Deadlock.		
DT2.R1. The system components should be resistant.	P	**DT2.C1.P.** Non-blocking synchronization algorithm [56].
		DT2.C2.P. Standard APIs to implement locking mechanism [56] (e.g., optimistic locking in a REST API).
DT3. Service Flooding.		
DT3.R1. The system should define limitations to the user provided data input.	P	**DT3.C1.P.** Setting a maximum password that can be processed [79].
DT3.R2. The system should allow income traffic communications from the authorized sources.	P	**DT3.C2.P.** White-listening source addresses [65].
		DT3.C3.P. Router access control lists and firewall rules [65]
DT4. HTTP Flooding.		
DT4.R1. The system should balance incoming network traffic.	C	**DT4.C1.P.** Load balancing algorithms [56] (e.g., Round Robin, Hash, IP Hash).
		DT4.C2.P. Load balancer software.

V. Security Requirements (Continued)

Security Requirement	Class	Security Control Components
ET1. Command Injection.		
ET1.R1. The system should conduct input data validation.	C	**ET1.C1.C.** Server-side validation [58]. **ET1.C2.C.** Client-side validation [58]
ET1.R2. The system should exploit security frameworks and libraries to validate input data before its usage.	D	**ET1.C3.D.** Hibernate Validator for Java [58].
		ET1.C4.D. Data Transfer Objects (DTOs) as an alternative for binding HTTP requests input parameters to server-side objects directly [58].
		ET1.C5.D. Query parameterization [58].
ET1.R2. The organization security personnel should conduct static code analysis before the system launch.	P	**ET1.C6.P.** Data-Flow Analysis [81] (e.g., FindBugs, Pixy).
		ET1.C7.P. Model checking [81] (e.g., QED [82]).
ET1.R3. The organization security personnel should conduct dynamic program analysis for the launched system.	P	**ET1.C8.P.** Whitelisting [81](e.g., SWAP, XSS-GUARD, DIDAFIT, SQLGuard).
		ET1.C9.P. Instruction set randomization [81].
		ET1.C10.P. Runtime tainting [81].
ET2. Bypass API Authentication.		
ET2.R1. The system should detect compromising the system boundaries.	D	**ET2.C1.D.** Network-based Intrusion Detection Systems (IDS) sensors [63, 78].
ET3. Session Hijacking		
ET3.R1. The system should protect the entire web sessions.	P	**ET3.C1.P.** Encrypted HTTPS (TSL) protocols [79].
ET3.R2. The system should exploit expiration timeouts for sessions.	P	**ET3.C2.P.** Exploit expiration timeouts for every session - idle or absolute [79].
ET3.R3. The system should manage sessions securely.	P	**ET3.C3.P.** Binding the session ID to other user or client properties (e.g., IP address, client-based digital certificate, etc.) [79].
ET3.R4. The system should follow session management policies.	P	**ET3.C4.P.** Restriction of multiple simultaneous logons from the same user based on the provided client IP addresses [79].

V. Security Requirements (Continued)

Security Requirement	Class	Security Control Components
ET4. Credentials Brute Force.		
ET4.R1. The system should verify a user password compliance with the password policy.	P	**ET4.C1.P.** Minimum length of the passwords is 8 characters [56, 40].
		ET4.C2.P. Maximum length is 64 characters due to limitations in certain hashing algorithms [56, 40].
		ET4.C3.P. Comparing password against a list that contains the blacklist (i.e., values known to be commonly-used, expected, or compromised [40, 56]).
		ET4.C4.P. Password-strength meter [40].
ET4.R2. The system should generate one-time password (OTP) in conjunction with user-generated password.	P	**ET4.C5.P.** Authenticators that generate high entropy output [40].
		ET4.C6.P. Regenerating an OTP, that is based on a real-time clock, once every 2 minutes [40].
		ET4.C7.P. Accepting a given time-based OTP only once during the validity period [40].
ET4.R3. The system should conduct user authentication throttling.	C	**ET4.C8.C.** Limit consecutive failed authentication attempts an a single account to no more than 100 [40].
		ET4.C9.C. CAPTCHA test [40].
		ET4.C10.C. White-listening IP addresses for authentication requests [40].

VI. Security Requirements for Bolt's system

Security Threat -> Related Risks
Requirements
ST1. Identity spoofing -> SR1
1. **ST1.R1.1.** *In-Vehicle Controller* should authenticate Passenger. 2. **ST1.R1.2.** *In-Vehicle Controller* should allow one authentication per a Passenger. 3. **ST1.R2.1.** *In-Vehicle Controller* should store user credentials securely. 4. **ST1.R3.1.** *Service Provider System* components should separate users roles. 5. **ST1.R4.1.** *In-Vehicle Controller* should implement access control mechanism. 6. **ST1.R5.1.** The Web Client should log access attempts to its interfaces. 7. **ST1.R6.1.** Bolt should have defined user management policy. 8. **ST1.R6.2.** *Service Provider System* should implement user management policy measures. 9. **ST1.R7.1.** *Service Provider System* should use secure protocols for delivering access credentials to Passenger. 10. **ST1.R7.2.** *Service Provider System* should use secure protocols for obtaining credentials from Passenger.
TT1. Data Storage Manipulation -> TR1, TR2, TR3
11. **TT1.R1.1.** *Service Provider System* should follow data management policy. 12. **TT1.R1.1.** *Service Provider System* should verify integrity of data from data storage. 13. **TT1.R1.2.** *In-Vehicle Controller* should use a local backups of data storage required for each initiated ride. 14. **TT1.R2.1.** *Service Provider System* should authenticate data storage users. 15. **TT1.R3.1.** *Service Provider System* should communicate with City Map data storage via channel protected by encryption protocol. 16. **TT1.R4.1.** *Service Provider System* should authorize access for data storage users.
TT2. API Parameters Manipulation -> TR4
17. **TT2.R1.1.** The system's API should hide sensitive data during external communications. 18. **TT2.R2.1.** *Service Provider System* should follow quality policy. 19. **TT2.R3.1.** *Service Provider System* should validate input data.
TT3. Software Source Code Manipulation. -> TR5, TR6
20. **TT3.R1.1.** *Service Provider System* should remain integral after software updates. 21. **TT3.R2.1.** *Service Provider System* should execute only authorized programs. 22. **TT3.R3.1.** *Service Provider* should follow policy of external systems quality. 23. **TT3.R4.1.** *Service Provider System* should use a channel protected by encryption protocol for remote managing.

VI. Security Requirements for Bolt's system (Continued)

RT1. System Log Files Manipulation. -> RR1
24. **RT1.R1.** *In-Vehicle Controller* should create backups of tracked users activities during the ride. 25. **RT1.R2.** *Service Provider System* should ensure data backups protection. 26. **RT1.R3.** *Service Provider System* should conduct action logging on the system components. 27. **RT1.R4.** *Service Provider System* should maintain timestamps consistency of logs on system components. 28. **RT1.R5.** *Service Provider System* should have central log management control. 29. **RT1.R6.** The security personnel should analyse system logs according to the log management policy.
IT1. Device Screen Capturing -> IR1, IR2
30. **IT1.R1.1.** *Passenger UI Client* should protect displayed on the mobile device sensitive data. 31. **IT1.R2.1.** *Service Provider System* shall guide Passenger to set required configuration settings on the mobile devices used for interaction with the system.
IT2. Device Notification Manipulation -> IR3
32. **IT2.R1.1.** *Service Provider System* should protect sensitive data provided to the mobile app. 33. **IT2.R1.2.** *Mobile Client* should protect sensitive data used for app notification. 34. **IT2.R2.1.** *Mobile Client* shall guide Passenger to set required notification settings on the mobile devices used for interaction with the system.
IT3. Access Stored Application Data -> IR4
35. **IT3.R1.1.** *Passenger UI Client* should store locally only required for normal functioning sensitive data. 36. **IT3.R2.1.** *Passenger UI Client* should encrypt sensitive data stored on mobile devices. 37. **IT3.R3.1.** *Passenger UI Client* should authenticate access to the data stored on mobile device. 38. **IT3.R3.2.** *Passenger UI Client* should authorize access to the data stored on mobile device. 39. **IT3.R4.1.** *Passenger UI Client* should follow the cache policy for the web applications.
IT4. Man-in-the-Middle Attack. -> IR5
40. **IT4.R1.1.** *In-Vehicle Controller* should verify integrity of the received data. 41. **IT4.R1.2.** *Passenger UI Client* should verify integrity of the received data. 42. **IT4.R2.1.** *In-Vehicle Controller* should identify unauthorized connections to the local area network. 43. **IT4.R3.1.** *In-Vehicle Controller* should ensure the confidentiality of transmitted via *Transmission Channel* information. 43.a. **IT4.R3.1.** *Passenger UI Client* should ensure the confidentiality of transmitted via *Transmission Channel* information. 44. **IT4.R4.1.** The service provider who owns the vehicles should control physical access to *transmission channel* within organizational facilities. 45. **IT4.R5.1.** *In-Vehicle Controller* should authenticate Passenger's mobile device before establishing connection. 46. **IT4.R6.2.** *Service Provider System* should follow the wireless capabilities policies. 47. **IT4.R7.1.** *Passenger UI Client* should protect the web sessions for the ride.

VI. Security Requirements for Bolt's system (Continued)

DT1. Communication Channel Jamming -> DR1, DR2
48. **DT1.R1.1.** *Service Provider System* should have backup channels for communication between its components.
49. **DT1.R2.1.** *Service Provider System* should use anti-jamming techniques.
DT2. Forced Service Deadlock -> DR3
50. **DT2.R1.1.** *Service Provider System* should be resistant.
DT3. Service Flooding -> DR4
51. **DT3.R1.1.** *In-Vehicle Controller* should define limitations to the user provided data input. 52. **DT3.R2.1.** *Service Provider System* should allow income traffic communications from the authorized sources.
DT4. HTTP Flooding -> DR5
53. **DT4.R1.1.** *Service Provider System* should balance incoming network traffic.
ET1. Command Injection -> ER1
54. **ET1.R1.1.** *In-Vehicle Controller* should conduct input data validation. 55. **ET1.R2.1.** *In-Vehicle Controller* should exploit security frameworks and libraries to validate input data before its usage. 56. **ET1.R3.1.** Bolt's security personnel should conduct static code analysis before the system launch. 57. **ET1.R4.1.** Bolt's security personnel should conduct dynamic program analysis for the launched system.
ET2. Bypass API Authentication -> ER2
58. **ET2.R1.** *Service Provider System* should detect compromising the system boundaries.
ET3. Session Hijacking -> ER3
59. **ET3.R1.1.** *Passenger UI Client* should protect the entire web session. 60. **ET3.R2.1.** *Passenger UI Client* should exploit expiration timeouts for sessions. 61. **ET3.R3.1.** *Service Provider System* should manage sessions securely. 62. **ET3.R4.1.** *Service Provider System* should follow session management policies.
ET4. Credentials Brute Force -> ER4
63. **ET4.R1.1.** *Service Provider System* should verify compliance of user password with the password policy. 64. **ET4.R2.1.** *Service Provider System* should generate one-time password (OTP) in conjunction with user-generated password.

VII. Example of Privacy Policy Analysis

Table 8. Key GDPR model elements in Ride Fulfilment Process (As-Is) in Bolt [69]

Key element from the GDPR model	**As-Is process characteristics (Y - Yes, N - No)**
Data Subject	Passenger
Controller	CentralSystem
Processor	AVsystem
Personal Data	D2.RideDetails
Processing Tasks	4. Approve ride 5. Notify passenger 10. Process ride
Category of personal data	General (according to the Privacy Policy for Passengers [69] D2. Ride Details contains: name, phone number and email address; geographical location data, time of service and destination of the trip (history of trips))
Legal ground	Consent ("The prerequisite for the use of Bolt services is passengers agreeing to the processing of identification and geolocation data." [69])
Consent collection	Y
Consent agreement characteristics	Clear purpose - Y,Unambiguous - N, Affirmative Action - N, Specific - Y, Distinguishable - N, Withdrawable - N, Freely Given - Y
Required information is provided before data collection	Y
The required information provided to the data subject prior to data collection in a form of Privacy policy, and includes	Controller contact info - Y, DPO contact info - Y, Purpose of processing - Y, Legal basis - Y, Data recipients - Y, Storage period - Y, Right to access - Y, Right to rectify - Y, Right to erasure - Y, Right to portability - N, Right to withdraw consent - N, Right to lodge complaint - Y, Automated decision making - N
Security/Privacy attributes of the processing system	Confidentiality - N, Integrity - N, Availability - N, Resilient - N, Pseudonimity - N, Data Minimization - N, Redundancies - N, Tested - N
Data storage characteristics	Data Storage - Y, Storage Limitation - Y
Secure technologies implemented	None
Implementation of security / privacy standard	N
Processing record maintain	N

VIII. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Mariia Bakhtina**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

   reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

   Securing Passenger's Data in Autonomous Vehicles,

   supervised by Raimundas Matulevičius and Mari Seeba.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Mariia Bakhtina
14/05/2021