Information Security Risks Analysis and Assessment in the Passenger-Autonomous Vehicle Interaction

Mariia Bakhtina* and Raimundas Matulevičius

Institute of Computer Science, University of Tartu, Tartu, 51009, Estonia
{mariia.bakhtina, raimundas.matulevicius}@ut.ee

Received: December 19, 2021; Accepted: February 10, 2022; Published: March 31, 2022

Abstract

Technological advances, consumer demands for advanced automotive assistant systems, and systems connectivity make cyber-security an essential requirement for ride-hailing service providers. While the final goal of autonomous vehicles (AVs) is to enable driverless rides, ride-hailing companies and their users - passengers - are the main stakeholders of the autonomous vehicles systems. However, to the best of our knowledge, there are no methods that prescribe how to protect passengers' data and manage security risks in AVs. This paper aims to determine how passenger's data can be protected in autonomous vehicles. The paper presents an approach to security risk management in the Passenger-AV interaction based on the domain model for information systems security risk management (ISSRM). 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. Finally, we present an approach for the security risks and requirements assessment that facilitate defining a risk reduction strategy. The research is conducted as a case study in the lab settings. The findings 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), risk assessment, requirements prioritisation

1 Introduction

The emerging concept of autonomous driving is of high interest nowadays and is supported with investments from government, private firms, and research centres [1]. According to [2], a vehicle equipped with an automated driving system (ADS) with Level 4 or Level 5 automation is the one that can conduct dynamic driving tasks without human intervention for all trips within the operation area (if any). Hereinafter, we refer to such a vehicle with ADS able to conduct trips without a driver as an *autonomous vehicle* (AV).

While the concept of AV brings to ride-hailing companies new opportunities, new security challenges appear. To trust self-driving technology and let it become ubiquitous, AV end-users (i.e., passengers) need to be sure that their personal data is protected and can be accessed only by authorised entities with harmless intention. Additionally, a ride-hailing company that aims to integrate AV into their fleet, first,

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 13(1):87-111, Mar. 2022
DOI: *10.22667/JOWUA.2022.03.31.087*

*Corresponding author: Institute of Computer Science, University of Tartu, Narva mnt. 18, 51009, Tartu, Estonia, Tel: +372-737-5421, Web: https://cs.ut.ee/

has to ensure the security of data critical for the execution of the driving tasks and for delivering the expected user experience and safety on the roads [3].

Security failure in the AV system may cause vehicle damage, financial losses, disclosure of sensitive personal data, and road accidents [3]. Unfortunately, much attention is paid to making autonomous driving ready-to-use technology, while the urgency of security risk management is omitted [4]. Such a knowledge especially concerns the use cases of the autonomous vehicles by end-users. Meanwhile, the newly introduced ISO/SAE 21434 standard imposes "requirements for cybersecurity risk management regarding concept, product development, maintenance of electrical and electronic systems in road vehicles, including their components and interfaces" [5]. Therefore, for companies that use AVs it is mandatory to manage security risks not only in their internal information systems, but also its integration with AVs.

This study aims to investigate ***how information security risks in Passenger-Autonomous Vehicle interaction can be managed***. For this purpose, we conduct applied exploratory research that results in the proposed approach for defining a risks reduction strategy to manage security risks. The method applied to the scenario produces the risk reduction strategy. We argue that the defined strategy should be used as a baseline for protecting Passenger-AV interaction from malicious attacks by the service providers who integrate AVs with their information systems.

The scope of the research is limited to a Passenger–AV interaction enabled by the ride-hailing company. Such a scope limitation allow us to investigate how AV operates from a passenger's perspective, excluding other humans that can interact with the driving AV (e.g., pedestrian, system administrator). The study relies on the business process and the AV system architecture designed within the autonomous driving lab. The considered system is supposed to be used by a ride-hailing service provider to allow customers to use driverless ride-hailing services. The considered system incorporates (*i*) AV system that conducts dynamic driving tasks of a single-vehicle, and (*ii*) an information system (IS) that offers infotainment service to passengers and help them to set interaction with a vehicle (i.e. 'Central System'). The systems can be either managed by the same or different service providers.

This paper is an extension of the work reported in [4], where we have presented an approach of the security requirements elicitation based on the developed threat model for the Passenger-AV interaction scenario. In this paper additionally we conduct risk assessment and a prioritisation of security requirements that results in the risk reduction strategy development.

The remainder of this article is structured as follows: Sect. 2 describes the case while Sect. 3 reflects the background of the study. In Sect. 4 we discuss the followed research method. Sect. 5 presents results of security risks management of the researched AV ecosystem. Finally, Sect. 6 concludes the paper by the discussion of the results, work limitations, and provides directions for the future research.

2 Case Description

The Passenger-AV interaction occurs during the *Ride Fulfilment* process, which consists of three parts that deliver value to a Passenger – *Ride Initiation*, *Ride Execution* and *Ride Post-Processing*. As the baseline of the surveyed processes, we used a user interface prototype[1] [6] designed in the autonomous driving lab. The prototype aims to increase trust in autonomous vehicles. We depict the Ride Fulfilment business process using Business Process Modeling Notation (BPMN) language to capture the process from the business perspective and show the data flows within it. The process model is presented in Fig. 1. The analysis presented in this paper covers the second part of the described process – Ride Execution sub-process as only in this phase of *Ride Fulfilment* a passenger actively interacts with the AV.

[1] `https://usability-test.laadamaailm.ee/` Accessed 10 Dec 2021

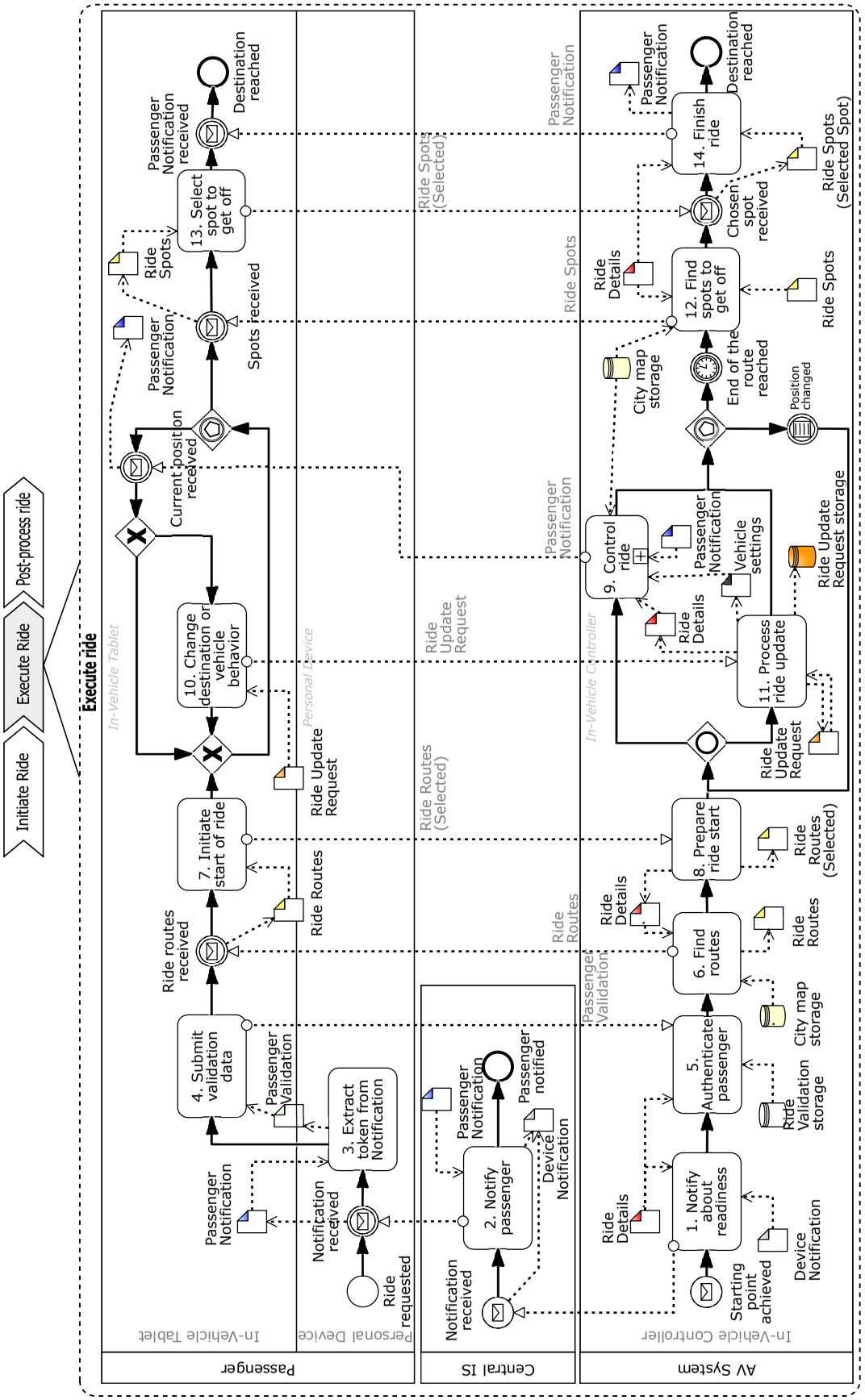

Figure 1: Ride Fulfilment business process [4]

The Ride Fulfilment starts when a *Passenger* initiates a ride by submitting a Ride Request in the Service Provider's App using a personal device. *Central IS* processes the request and sends it to the *AV System* of the assigned vehicle to execute the ride. When the AV achieved the starting point of the ride, *Passenger* authenticates themselves and initiates the ride start. Once the ride started, *AV system* controls the ride by executing dynamic driving tasks. Meanwhile, *AV system* informs *Passenger* about the current location, and *Passenger* can change the destination point or the vehicle behavior (e.g., speed) by making a request on the In-Vehicle Tablet on which the web client of the system is opened. When approaching the destination point, the vehicle asks *Passenger* to select a spot to get off among the available spots near the destination. As soon as the selected spot achieved, *Passenger* is notified and asked to leave the vehicle. Finally, *Central IS* finishes the ride by processing the captured during the ride data to improve future services.

3 Background

Assuring information security is an essential part of the system development lifecycle, which aims to support the quality of a developed system and the consequent acceptance of the system by its users. According to [7], 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." This section presents the fundamental security risk management concepts we use in our study. Additionally, we discuss the related work of the study to highlight the knowledge gap we aim to address.

3.1 Security Risk Management

While such standards as ISO/IEC 2700x series, National Institute of Standards and Technology (NIST) special publications generally guide security risk management, the introduced methods, perspectives, and terminologies of security risk management vary from one standard to another. 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 [8], [7] like OCTAVE, the NIST Cybersecurity Framework, MEHARI. The domain model for information systems security risk management (ISSRM) has been developed [9] to avoid misunderstanding between security experts and orchestrate standards mentioned above. According to the survey results [10], ISSRM was assessed as one of the most proficient concepts that implement ISO/IEC 27001 standard requirements. Security modelling languages support the ISSRM domain model [8], which helps cover the model's concepts using the corresponding tools. To analyse the selected case scenario, we are using the ISSRM domain model as the baseline.

According to the ISSRM domain model (see Fig. 2), there are three key groups of concepts. The *asset*-related concepts describe the organisation's assets, their value, and the reasoning why they should be protected. The *risk*-related concepts correspond to risk itself and its components. The *risk treatment*-related concepts describe how risks can be treated.

For the measurement of concepts in the ISSRM domain, a set of security metrics is defined. The business assets are characterised by *Value* metric. This metric describe the importance of the business asset to the operation of the business. The *security need* metric defines the necessity of keeping security criteria of the business assets. The risk is measured with the *risk level*. While a risk is composed of risk event and its impact on the assets, the risk level depends on the *risk event potentiality* and *impact level* which depends on the value of harmed business assets. In turn, the risk event potentiality is determined based on the *threat likelihood* and *vulnerability level*. As a threat is composed of a threat agent and

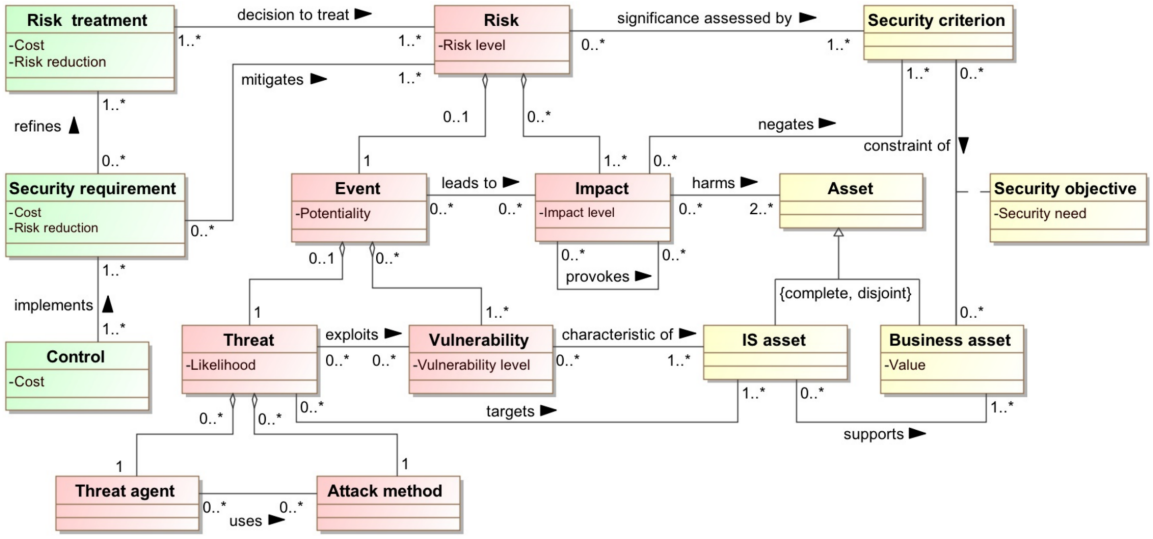

Figure 2: The ISSRM domain model [7]: yellow entities represent asset-related concepts, red entities - risk-related concepts, and green entities - treatment-related concepts

employed attack method, the threat likelihood is estimated using its components metrics which may depend on one another. Finally, there are two metrics to measure a security requirement, namely the *cost* which is defined by the *cost* of controls and *risk reduction level* that estimate the risk mitigation impact of the requirement.

3.2 Related Work

Numerous studies have attempted to address information security and correspondent risk management in autonomous vehicles looking at the problem from different perspectives. Most of the researches, like [11], [12] and [13], are focusing on the security of in-vehicle components, vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication. While AV-Human communication is mostly considered as interaction with pedestrians, the passenger's role is not widely discussed as well as the role of external service providers (e.g., ride-hailing service or ride-sharing).

In [13], authors have discussed security and privacy threats in the case of autonomous and cooperative automated vehicles (CAVs) covering In-Vehicle, V2I, and V2V interaction. The study highlighted some attacks which can take place in the Passenger-AV interaction such as attacks targeting in-vehicle devices (e.g., hand-held devices connected to the infotainment system via USB, Wi-Fi or Bluetooth), other electronic devices and maps (used by a vehicle in case of non-real-time detection of the road). Mostly, 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. Thus, 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 [11] proposed a comprehensive taxonomy of AV attacks and defenses that assist AV system architectures development. In [14] authors emphasize 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, ... and what potential risks there are." In [15] authors presented a taxonomy of threats and generalized attack surfaces for CAV applications. In the same paper, the role of the vehicle's components security within the CAVs supply chain was highlighted. For example, testing, anti-malware updates, and phys-

ical access to the AV components by original equipment manufacturers and vendors are recommended to be logged to preserve the security of the whole vehicle system and users' privacy. Additionally, human factor in the safety and security of CAVs was discussed. Our work is complimentary to the studies in [14, 11, 15] as we study the scenario that covers security attacks which target vehicular network, vehicle system components and passenger's device, and propose measures to address corresponding security risks. The current work also 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 worth mentioning. The European Union Agency for Cybersecurity (ENISA) project [3] 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, the guide [16], provided by Information-technology Promotion Agency (IPA), Japan, overviews the potential threats to autonomous vehicles on the high level of abstraction. It gives the general recommendations regarding security efforts in phases of automotive systems' lifecycle, which are not scenario oriented, but rather system functionality focused. However, they consider a passenger as a passive system user, which only obtains information from the infotainment system. In [17] 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' interaction with a vehicle was presented. Therefore, this work aims to build an analogue guideline with less focus on system functionality for the AV system developers and service-providers which use enable active Passenger-AV interaction.

In the latest report [18] ENISA highlighted the need for security risk management over the products and services lifecycles in the sector of connected and automated mobility, which includes an ecosystem of services, operations and infrastructure autonomous and connected vehicles. They also recommend conducting threat modelling for revealing relevant threat scenarios and address security issues in the early stages of system development. Additionally, the high level of the interconnectivity of the ecosystem components means that lack of security protection may lead to compromising the system at the scale of a fleet of vehicles. Thus, this paper complements the high-level recommendations in [18] as we propose the security risks management approach to the scenario where the perimeters of ecosystem components intersect so that security measures and risk management should be considered holistically for preserving AV ecosystem protection.

To sum up, previous works discussed either general attack vectors on autonomous vehicle systems and defenses rather than examples of their applicability for the systems, or the general guides of security risk management of vehicular system, while none of the studies comprises a comprehensive overview of managing information security risks on a scenario level. Therefore, this paper aims to illustrate the the first stages of information risk management by incorporating a more technical and detailed attack methods discovery, and higher-level risk management approaches.

4 Research Method

This paper presents the applied exploratory research of Passenger-AV interaction. The aim of this study is to investigate *how to manage information security risks in Passenger-Autonomous Vehicle interaction*. The ISSRM domain is used as the guide for security risk and requirements definitions. Therefore, to address the goal of the study, we aim to answer four following sub-question by following the process depicted in Fig. 3. To analyse the given scenario, the research process embraces two parallel sub-processes: (i) theoretical artefact development based on the literature review and (ii) the case analysis by applying the derived artefacts.

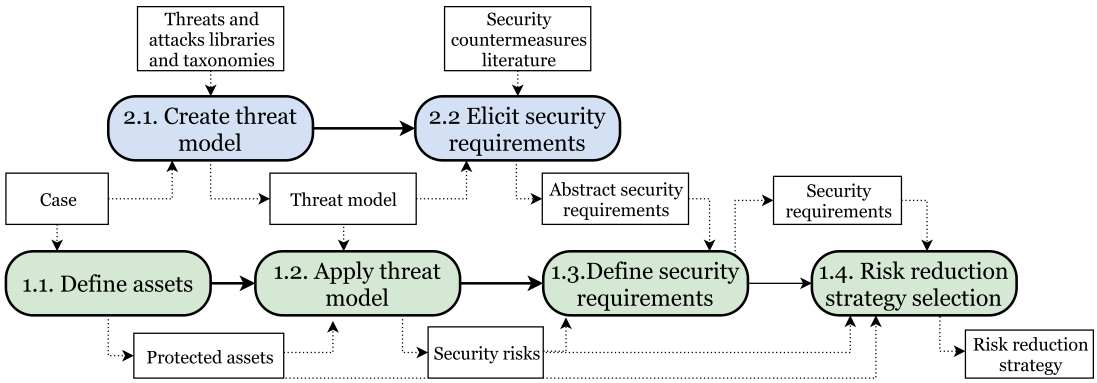

Figure 3: The research conduction map

RQ$_1$: *What assets should be protected in the Passenger–AV interaction?* First, we conduct the activity *1.1.* to answer *RQ*$_1$ by defining the assets that should be protected based on the designed case. The business assets are derived based on the given interaction scenario and the data structure. To this end, the UML class diagram is created to identify the critical data entities and interdependencies. The combination of the flows depicted in the business process diagram and the data structure enables us to identify which assets should be protected, which security criteria assured and how assets are connected (see Sec. 5.1).

RQ$_2$: *What are the security risks in the Passenger–AV interaction?* To answer *RQ*$_2$, we follow the threat-driven approach for defining security risks in the given interaction scenario. For this purpose, we review attack libraries, threats and vulnerabilities taxonomies and the threat modelling framework to build a threat model on the step *2.1* (see Sect. 5.2). The application of the threat model to the derived assets (activity *1.2*) allows us to identify a set of security risks in the Passenger-AV interaction scenario.

RQ$_3$: *What are the security requirements to mitigate security threats in the Passenger–AV interaction?* By answering RQ$_3$, we aim to define the countermeasure to reduce the security risks. The countermeasures are formulated in the form of security requirements that a ride-hailing company can implement. The activity *2.2* corresponds to the security requirements elicitation based on the literature review. After, the abstract security requirements are applied instantiated in the context of the scenario (activity *1.3*).

RQ$_4$: *What is the security risk reduction strategy?* Finally, having security requirements from RQ$_3$, we aim to define which security countermeasures should be implemented primarily in the system. To answer RQ$_4$, we prioritise requirements by finding a trade-off between their impact on the risks, cost and protected asset's value. For that, we propose to assess the security risks levels before and after implementation of security countermeasures and prioritise requirements based on the risk assessment results and value-cost assessment of countermeasures. Based on the requirements prioritisation results, we propose the risk reduction strategy.

5 Security Risk Management of the Passenger-AV interaction

In this section, we present the results of security risks analysis for the presented in Section 2 scenario according to the research questions. First, we introduce construction of the threat model and security risks. The possible security measures to address risks are presented in the form of security requirements. Finally, we conduct risk assessment to prioritise the countermeasures based on the risk reduction level, assets values and costs. In this section, we also introduce to readers the background about threat mod-

elling, including common threats taxonomies and libraries from which a threat model for Passenger-AV interaction is developed.

5.1 Protected Assets

This section answers the RQ_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 need of a system. Confidentiality, integrity, and availability, also known as CIA triad, forms the main security criteria which can characterise business assets.

The general data structure of the system is presented in Fig. 4 in the form of a UML class diagram. The identified business assets are illustrated on the diagram as green entities, and system assets are depicted as red entities.

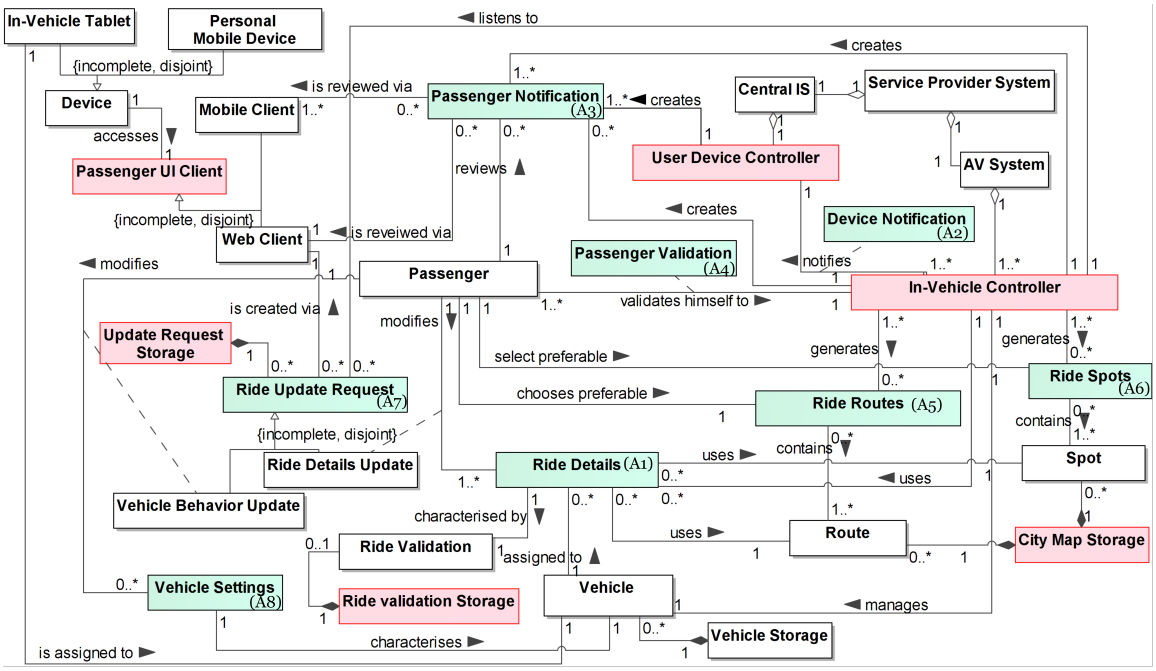

Figure 4: Data structure of the system [19] (green entities - business assets; red entities - system assets)

Ride Details (A1) 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, an involved in the ride *vehicle*. Also, it contains the reference to the entity, which corresponds to *Passenger* and stores her personal data, e.g. payment details. Meanwhile, *Passenger Validation* asset (A4) contains credentials which a Passenger should use for starting a dialog with the system, and, consequently, start the ride. As a result, the *Ride Details* asset aggregates all the other assets to enable Passenger conduct supervisory control over the AV during Ride Fulfilment.

Now let us describe how the system assets are organised and what are their key functionality. A service provider system consists of the two main components: Central IS 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 corresponds to the front-end part 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 is application programming interfaces (APIs), which facilitate communication between the system components - *Central IS API* and *AV System API*.

5.2 Threat Modelling

According to [20] and [7], information security risks are mostly defined by the attacks that an adversary employs to target a system assets. Thus, threat-driven approach for risks identification is commonly used [21, 20] for guiding the ISRM. The threat model should be defined as a primary step for risks identification. The primary deliverable of this subsection is a threat model for the Passenger-AV interaction scenario defined in Section 2.

5.2.1 Supporting resources

As the supportive tools and resources of common threats, attack vectors and techniques, we have selected four sources: (i) Common Attack Pattern Enumeration and Classification (*CAPEC*) [22]; (ii) the *STRIDE* approach [23]; (iii) Adversarial Tactics, Techniques, and Common Knowledge (*ATT&CK*) framework [24]; and (iv) the list of vulnerabilities provided by the Open Web Application Security Project (*OWASP*) [25].

The *STRIDE* approach [23] is designed for eliciting system security threats. STRIDE is supposed to be used at the beginning of ISRM during the defining potential risks and attack vectors. The Common Attack Pattern Enumeration and Classification [22] (*CAPEC*) is a comprehensive, community-created catalog of attack patterns. It defines the informal taxonomy of attack-pattern classes and provides the formal description of each attack class. The taxonomy is organised hierarchically based on its domain and mechanisms of attack specifying the vulnerabilities it addresses. CAPEC is supported by references to the targeted vulnerabilities and possible mitigations. Adversarial Tactics, Techniques, and Common Knowledge (*ATT&CK*) framework [24] is a knowledge base of adversarial techniques which helps to classify attacker's actions for different platforms (e.g., Windows, Android). It is focused on techniques in the context of tactics an adversary wants to apply to attack a specific component or endpoint. Concrete procedure examples support each technique an adversary may use, system requirements for implementing the tactics, possible detection methods, and mitigations. The techniques are mapped to the corresponding attack patterns. Another resource for the threat model creation is the list of vulnerabilities provided by the Open Web Application Security Project (*OWASP*) [25] is considered a starting point for developing secure software focused on defensive mechanisms and controls. The approach does not consider the prospect of a threat agent or any application implementation details. Thus, for each specific case, a threat agent, assets, and corresponding impact should be considered besides, respectively.

The selected resources were chosen due a number of reasons. First, the 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. Second, the threats and attacks within the repositories are described in details, illustrated by real case implementations and the attacks supported by high-level mitigations. Third, the classified attacks and threats are relevant for the researched system architecture and process. And, finally, the combination of the repositories, taxonomies enable to derive threats for the system starting from the general attack vectors until the concrete threats covering the full scenario.

5.2.2 The threat model for Passenger-AV interaction

Fig. 5 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. Each threat is supported by the reference to the source it was elaborated from. The detailed description of the threats (targeted vulnerability, threats agent, attack method, and potential impact) can be found in [19].

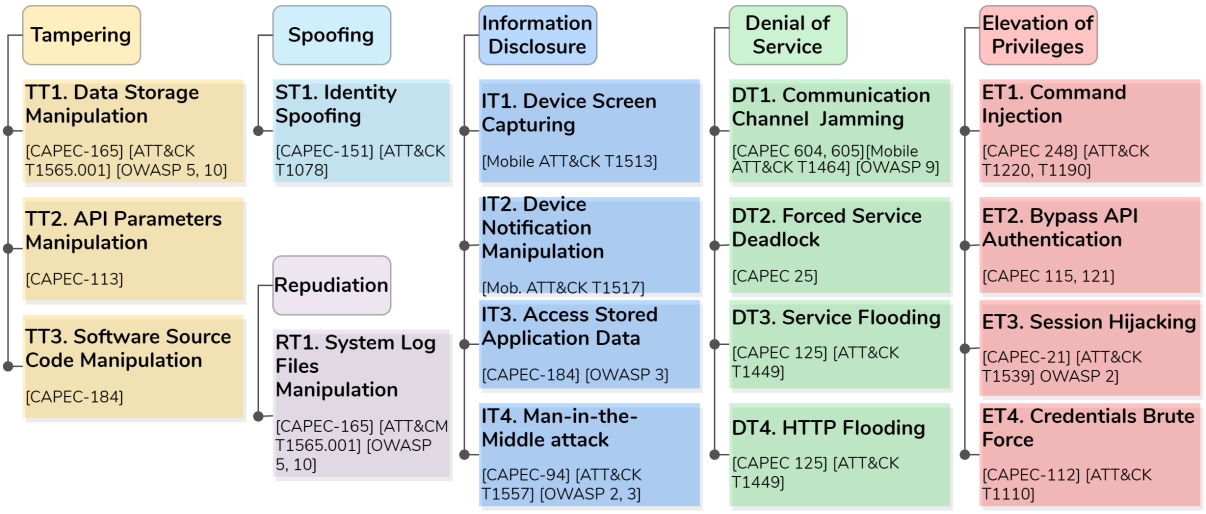

Figure 5: The threat model for the Passenger–AV interaction [4]

Spoofing refers to identity spoofing attacks where an attacker pretends to be a legitimate passenger. To violate the system's *authentication* mechanism, an attacker uses the obtained credentials (ST1). In the case of ***tampering***, an attacker intentionally modifies a system, network, its behavior, or the data to violate their *integrity*. These threats target data storage (TT1) and software source code files (TT3), which are used during the ride execution and are critical to the 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 are targeting 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, 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 *authorization*. 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). It should be noted that the identified threats are interconnected as implementation of one of them enables execution of another.

For example, successfully implemented *IT4. Man-in-the-Middle attack* enables execution of *TT2. API Parameters Manipulation* which in turn may result in *DT2. Forced Service Deadlock*.

5.3 Security Risks Identification

The current subsection answers RQ_2 by identifying security risks based on the developed threat model for Passenger-AV interaction. The risk model for the observed scenario can be derived by instantiating attacks from the threat model to the business assets and its vulnerabilities. As a result, for the assets identified in Sec. 5.1, the risk model includes 22 information security risks that can take place in the Service Provider System. The complete model can be found in [19]. 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 includes the harm to the system components, which as a result may result in getting access to any sensitive data which is visible to the system. Table 1 contains the full list with 22 risks identified for the Passenger-AV interaction scenario.

Table 1: Security risks in the Passenger-AV interaction scenario

RiskId.Name [*Description ThreatId → Affected AssetId*]		***RiskId.Name*** [*Description ThreatId → Affected AssetId*]	
SR1. Passenger Identity Spoofing		**TR1. City Map Storage Manipulation**	
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.	ST1 → A1, A3, A7	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.	TT1 → A5, A6
TR2. Ride Validation Storage Manipulation		**TR3. Ride Update Request Storage Manipulation**	
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.	TT1 → A4	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).	TT1 → A7
TR4. Central IS API Parameters Manipulation		**TR5. User Device Controller Source Code Manipulation**	
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.	TT2 → A2-3	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.	TT3 → A2-3
TR6. In-Vehicle Controller Source Code Manipulation		**RR1: System Log Files Manipulation**	
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.	TT3 → A1, A3, A5-8	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.	RT1 → A1
IR1. Personal Mobile Device Screen Capturing		**IR2. In-Vehicle Tablet Device Screen Capturing**	
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.	IT1 → A3	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.	IT1 → A3

Table 1: Security risks in the Passenger-AV interaction scenario (Continued)

IR3. Personal Mobile Device Notification Manipulation		**IR4. Access Stored Application Data**	
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.	IT2 → A3	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.	IT3 → A3
IR5. Man-in-the-Middle		**DR1. Front-Back Communication Channel Jamming**	
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.	IT4 → A3	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.	DT1 → A3-7
DR2. Back-Ends Communication Channel Jamming		**DR3. Forced In-Vehicle Controller Deadlock**	
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	DT1 → A2	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.	DT2 → A5-6
DR4. Service Flooding		**DR5. HTTP Flooding**	
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	DT3 → A7	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.	DT4 → A4
ER1. Command Injection		**ER2. Bypass API Authentication**	
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.	ET1 → A4	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.	ET2 → A7
ER3. Passenger Session Hijacking		**ER4. Passenger Credentials Brute Force**	
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.	ET3 → A7	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.	ET4 → A3

To illustrate the attack implementation, we are using the security extension to BPMN [26], which supports the ISSRM domain model. Fig. 6 contains an example of the derived security risks – namely, IR5 the Man-in-the-Middle (MitM) attack execution which targets *Passenger Notification*. According to CAPEC, the MitM attack required medium skills level required, but has high impact, as it enables an adversary to conduct further attacks on the system. In Fig. 6, we see an attacker as an additional entity that intercepts in the transmission channel aiming to define when the AV with the Passenger reaches the desired place on their route.

The implementation of Man-in-the-Middle attack (threat IT4) primarily aims to negate the confidentiality of Passenger Notification. However, the effective delivery enables an attacker to conduct a set of further attacks that already may target the vehicle's functions, which may provoke the loss of passenger's safety.

Fig. 7 illustrates implementation of another risk by an attacker - TR6. In-Vehicle Controller source

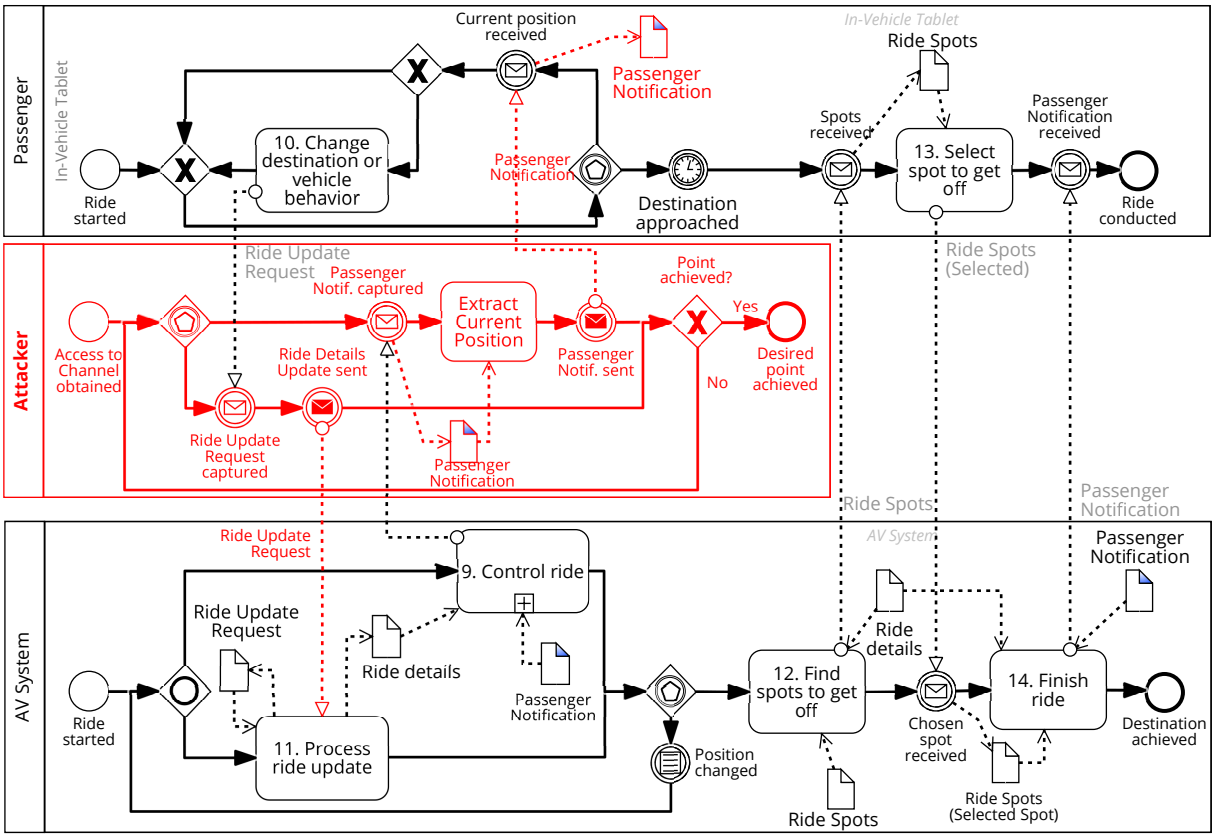

Figure 6: Execution of the risk IR5: Man-in-the-Middle attack [4]

code manipulation. While the threat TT3, on which the risk is based, has quite many pre-requisites, the successful execution may let an attacker change the functionality of the In-Vehicle Controller that plays the key role in the ride execution. This risk has a relatively high impact on the AV system and the interaction scenario as it may affect all protected assets to which the Controller has access. Depending on the malicious intentions, an attacker may update the assets' values (e.g., change the destination of the ride or passenger's payment details to cause financial losses), get access to the sensitive data (e.g., passenger's location) or make some assets not available (e.g., by deleting the Ride Details' assets that may prevent AV from the ride execution).

5.4 Security Requirements Elicitation

The current subsection contains a list of the derived security requirements that answers the RQ_3.

5.4.1 Security requirements elicitation

According to [27] and [7], 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 [28], National Highway Traffic Safety Administration (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[29]). Similarly, the report by ENISA [3] contains a set of good practices for smart cars. It stresses that for conducting information

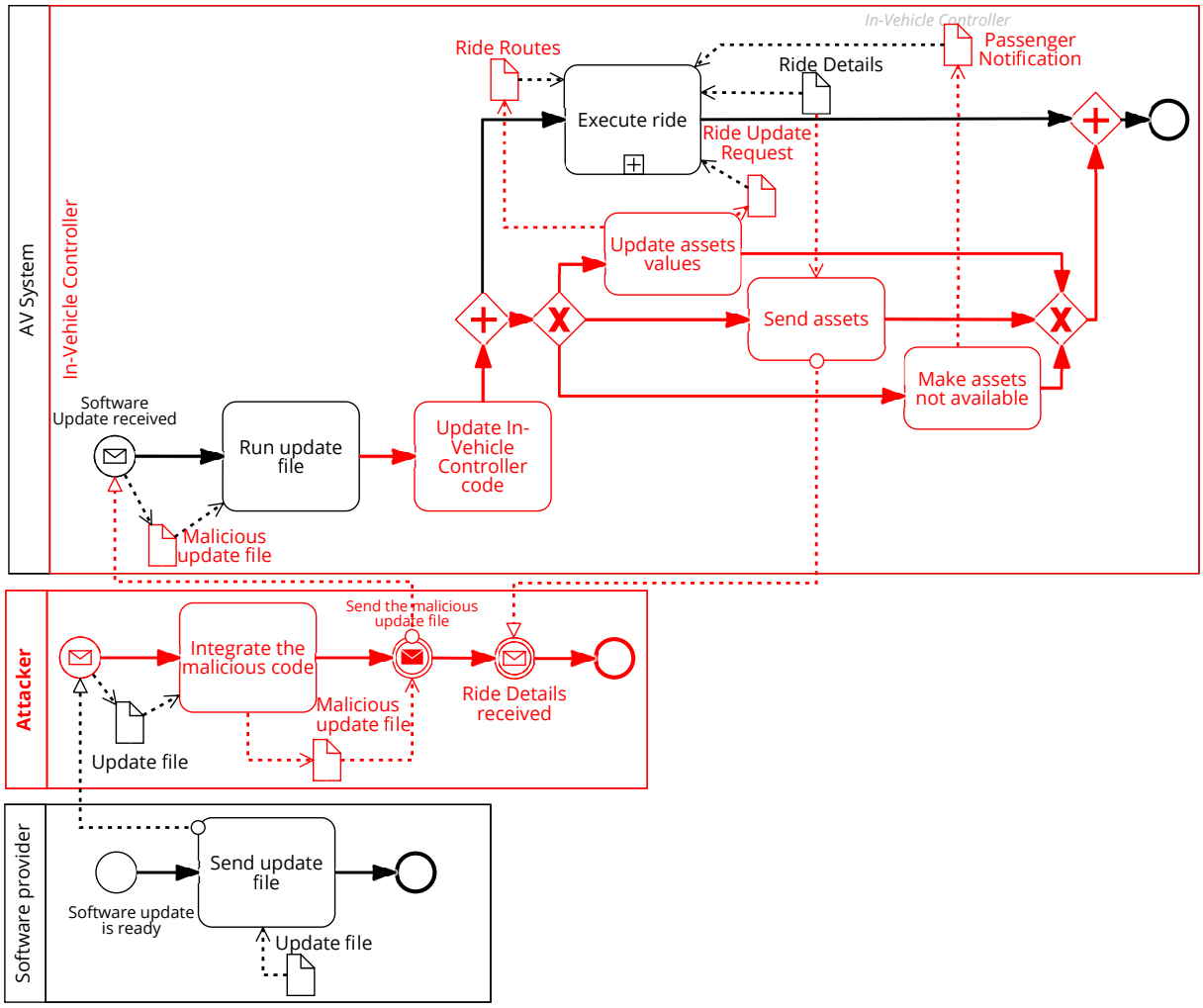

Figure 7: Execution of the risk TR6: In-Vehicle Controller source code manipulation

security management, it is important to use aforementioned standards along with SAE J3061 [30] and NIST 800-53[31]. IPA proposes the guide [16] for achieving a security level in the automotive systems. They highlight the 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.

Meanwhile, threat-driven requirements elicitation approach supports security requirements categorisation: (i) *preventive*; (ii) *detective*; (iii) *corrective*. The analogue taxonomy of security countermeasures for the AV defence is presented in [11]. Such requirements categorisation enables their prioritisation based on the impact of the risks. For example, a higher priority could be given to preventive requirements that preserve threats that enable further attacks execution.

5.4.2 Defined security requirements

For mitigating the defined risks in the Passenger-AV scenario, we considered the security controls from the aforementioned standards and libraries. The requirements were later defined using the inductive approach from the found controls. Using [25], [22], [29], [31], [24] as the main primary sources, we have elicited 56 requirements which are supported with the 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 [19], while Table 2 represents security countermeasures which are supposed to address risks illustrated in Sec. 5.3 (namely, IR5 and TR6).

Table 2: Security countermeasures identification (P - preventive, D - detective, C - corrective)

Security Requirements	Class	Security Control Components	Security Requirements	Class	Security Control Components
IT4. Man-in-the-Middle Attack			**TT3. Software Source Code Manipulation**		
IT4.R1. The system should verify integrity of the transmitted data.	D	**IT4.C1.** Cryptographic hash functions: message authentication code (MAC) algorithms [11], [32], digital signature, and checksums [32].	**TT3.R1** The system should remain integral after software updates.	P	**TT3.C1.P.** Software updates validation [22].
IT4.R3. The system should ensure the confidentiality of transmitted information.	P	**IT4.C4.** Cryptographic mechanisms: SSL/TLS protocol [29], IPSec protocol suite [32]. **IT4.C5.** Advanced encryption standard (AES) to encrypt wireless data in transit [29].	**TT3.R2.** The system should execute only authorized programs.	P	**TT3.C2.P.** Application control for regulating external files execution [24]. **TT3.C3.P.** Application whitelisting technology [29].
IT4.R5. The system should authenticate device before establishing connection.	P	**IT4.C9.** Bidirectional cryptographically based authentication [32].	**TT3.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 [29]).
IT4.R6. The system should follow the wireless capabilities policies.	P	**IT4.C10.** Usage of wireless networking capabilities only for essential functions [29], [32].			

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 lies in the interception of system security and human behavior management.

Fig. 8 illustrates how the risk TR6 (In-Vehicle Controller source code manipulation) can be reduced with the TT3.R1 requirement. To remain integral after the software updates, the AV system could implement the requirement in the following way: first, it checks the integrity of the received update files to check its author, then the received updates can be applied and the integrity of the updated system can be checked by running tests (e.g., unit or integration), and finally, in case of found issues, the system should rolls back the update, and after – proceed with the ride execution.

5.5 Risks Mitigation Impact Assessment

Following the ISSRM domain model, the risk is measured with the risk level metric, which is estimated based on metrics of the risk's components. So having identified assets, risks and requirements, we assess the risk mitigation impact to recommend the risk reduction strategy that answers RQ_4.

5.5.1 Asset-related concepts assessment

Estimate business asset value. First, we estimate asset value AsV. For this reason, we consider four impact areas [33] to which an asset can potentially contributes: (IA_1) healthy and safety, (IA_2) human right and privacy, (IA_3) business activities and delivered service, (IA_4) customer experience. We estimate the impact of each area IA_i for each asset using a scale from 0 to 3 (Very low to High impact), and the total asset value is composed of the impact areas' values: $AsV = \sum IA_i$, where IA_i are the impact area values, $i \in [1,4]$.

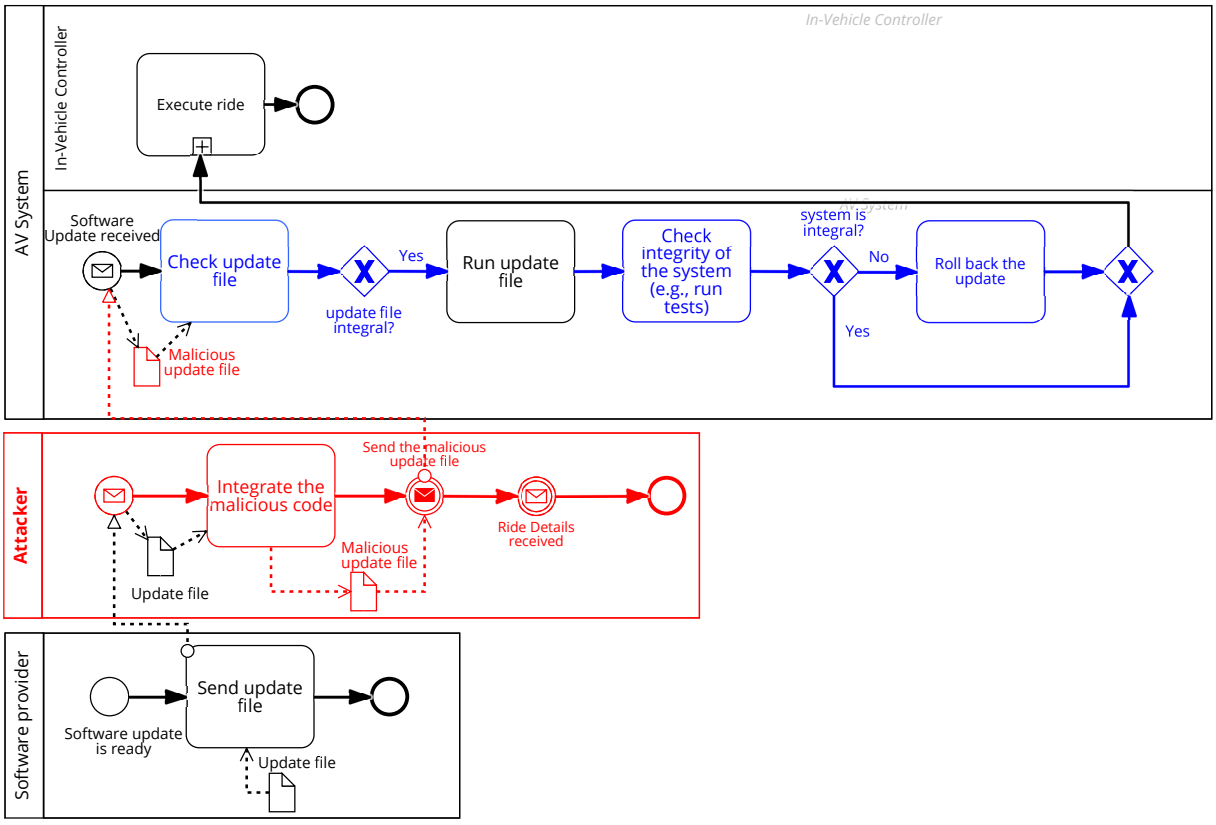

Figure 8: The risk TR6 reduction with the TT3.R1 requirement

The impact values of different areas are the subjective estimation of the assets based on the business objectives, system architecture (see Sec. 5.1), and system's usage scenario (see Sec. 2). For example, the value of Passenger Notification asset is estimated as equal to 6 due to the following assessment of impact areas: $IA_1 = 0$ as the asset presence does not affect the safety or health of the passenger; $IA_2 = 1$ as even though the asset contains passenger's location during the ride (sensitive personal data) the absence of the asset does not threaten passenger's privacy; $IA_3 = 3$ as the continuous supply of the asset is required to let the passenger conduct supervisory control over the vehicle, the delivered to the customer service highly depends on the asset; $IA_4 = 2$ as having the asset in place increases the level of customer's experience by giving them information about the current ride status.

5.5.2 Risk-related concepts assessment

Assess vulnerability level. The vulnerability level V characterises the likelihood of having the vulnerability, the potential damage of having the vulnerability and the urgency to address it. The vulnerabilities of system assets are assessed by experts taking into account ratings from OWASP Top Ten [25] and Common Weakness Enumeration [34] database. While a threat can exploit more than one vulnerability, the total vulnerability level in the risk VL is composed of all the exploited vulnerabilities: $VL = \sum V_i$, where V_i are levels of vulnerabilities addressed in the risk.

Estimate threat likelihood. To estimate the likelihood of threat ThL we assess such parameters of attack method and attacker as (i) required resources R to exert a threat and (ii) required know-how K to post a threat [35]. For the first parameter, we use a scale from 0 (no additional tools required) to 3 (ad-

vanced tools needed), and for the second parameter, we use a scale from 0 (no prior knowledge required to use the technique as a black-box) to 2 (domain knowledge for white-box approach implementation). For assessment of the threat likelihood based on the R and K, the following calculations are done:

$$ThL = \begin{cases} 3, if\ R+K \in \{0,1\} \\ 2, if\ R+K = 2 \\ 1, if\ R+K \in \{3, 4\} \\ 0, if\ R+K = 5 \end{cases}$$

Security experts estimate the assessment of required resources R and know-how K based on the description of threats, attacks and prerequisites to attackers [22, 24, 25]. For instance, the likelihood of the threat ET4 (Credential Brute Force) is high (*ThL*=3) as to exert if and the attacker does not need to have any knowledge about the system or technical background (*K*=0) and only the publicly available brute force tools (*R*=1) are required.

Determine threat event potentiality. Risk event potentiality *EP* is calculated based on the metrics of a threat and exploited vulnerabilities that form a risk event. So, the potentiality of the risks event depends on the threat likelihood and the total level of the exploited vulnerabilities: $EP = ThL + VL$.

Determine risk impact. The risk impact *I* of the risk considers the value of affected assets and the violated security criteria of each business asset. Therefore, the risk impact is calculated as following: $I = \max(AsV_i * n_i)$, where AsV_i is the asset value, n_i is the number of violated security criteria for the asset *i*.

Assess risk level. Having the metrics of all the risk components determined, we perform the following calculation to assess the risk level *R*: $R = EP * I$. Thereby, the risk level considers the potentiality of the risk event and the impact on the business assets it may cause. By this point, the risk assessment activity is finished. Moreover, as stated in [35], the security risks of high level should be considered as candidates for the safety risks in the context of automotive systems under development which is the autonomous vehicle system in our case.

The assessment results reveal that risks DR2 (Back-Ends Communication Channel Jamming) and IR4 (Access Stored Application Data) have the lowest risk levels (8 and 12 respectively), while risks TR6 (In-Vehicle Controller Source Code Manipulation), TR5 (User Device Controller Source Code Manipulation) and IR3 (Personal Mobile Device Notification Manipulation) have the highest risk levels (72, 54 and 48). The results of risk assessment can be found in [36], whereas Table 3 demonstrates examples of a few risks assessment results.

Table 3: Example of risk metrics before and after implementation of the security requirements (the detailed metrics calculations cab be found in [36])

Risk (before treatment)						Risk (after treatment)						Risk Reduction Level RRL	Business Asset Value	Cost of require-ment	RRL-Value	RRL-Cost	Value-Cost	Final priority	
Risk ID	Vulner. Level VL	Threat Likelihood ThL	Risk Event Potential. EP	Impact level	Risk Level	RiskID. ThreatID.Reqld	Vulner. Level VL	Threat Likelihood ThL	Risk Event Potential. EP	Impact level	Risk Level				Graph 1	Graph 2	Graph 3		
SR1	1	3	4	8	32	SR1.ST1.R1	1	0	1	8	8	**24**	**8**	**180,000**	2	1	2	**5**	Medium
						SR1.ST1.R2	1	1	2	8	16	**16**	**8**	**9,576**	2	2	3	**7**	High
						SR1.ST1.R3	1	3	4	8	32	**0**	**8**	**14,364**	2	2	3	**7**	High
						SR1.ST1.R4	1	3	4	8	32	**0**	**8**	**33,516**	2	1	2	**5**	Medium
						SR1.ST1.R5	0	2	2	8	16	**16**	**8**	**4,788**	2	2	3	**7**	High
						SR1.ST1.R6	1	2	3	8	24	**8**	**8**	**14,364**	2	2	3	**7**	High
						SR1.ST1.R7	1	1	2	8	16	**16**	**8**	**14,364**	2	2	3	**7**	High
TR1	6	0	6	5	30	TR1.TT1.R1	3	0	3	5	15	**15**	**5**	**19,152**	1	2	2	**5**	Medium
						TR1.TT1.R2	4	0	4	5	20	**10**	**5**	**9,576**	1	2	2	**5**	Medium
						TR1.TT1.R3	6	0	6	5	30	**0**	**5**	**14,364**	1	2	2	**5**	Medium
						TR1.TT1.R4	6	0	6	5	30	**0**	**5**	**9,576**	1	2	2	**5**	Medium
TR5	3	0	3	18	54	TR5.TT3.R1	1	0	1	0	0	**54**	**6**	**9,576**	2	3	2	**7**	High
						TR5.TT3.R2	2	0	2	0	0	**54**	**6**	**9,606**	2	3	2	**7**	High
						TR5.TT3.R3	1	0	1	6	6	**48**	**6**	**2,394**	2	3	2	**7**	High
						TR5.TT3.R4	3	0	3	18	54	**0**	**6**	**9,576**	1	2	2	**5**	Medium
TR6	3	0	3	24	72	TR6.TT3.R1	1	0	1	0	0	**72**	**8**	**9,576**	3	3	3	**9**	High
						TR6.TT3.R2	2	0	2	0	0	**72**	**8**	**9,606**	3	3	3	**9**	High
						TR6.TT3.R3	1	0	1	8	8	**64**	**8**	**2,394**	3	3	3	**9**	High
						TR6.TT3.R4	3	0	3	24	72	**0**	**8**	**9,576**	2	2	3	**7**	High
RR1	4	1	5	8	40	RR1.RT1.R1	4	1	5	0	0	**40**	**8**	**4,788**	3	3	3	**9**	High
						RR1.RT1.R2	4	1	5	0	0	**40**	**8**	**57,456**	3	2	2	**7**	High
						RR1.RT1.R3	2	1	3	8	24	**16**	**8**	**1,197**	2	2	3	**7**	High
						RR1.RT1.R4	4	1	5	8	40	**0**	**8**	**48**	2	2	3	**7**	High
						RR1.RT1.R5	2	1	3	8	24	**16**	**8**	**33,576**	2	1	2	**5**	Medium
						RR1.RT1.R6	3	1	4	8	32	**8**	**8**	**9,576**	2	2	3	**7**	High
IR5	4	1	5	6	30	IR5.IT4.R1	2	1	3	6	18	**12**	**6**	**19,152**	1	2	2	**5**	Medium
						IR5.IT4.R2	3	0	3	0	0	**30**	**6**	**14,364**	2	2	2	**6**	Medium
						IR5.IT4.R3	2	1	3	0	0	**30**	**6**	**14,364**	2	2	2	**6**	Medium
						IR5.IT4.R4	1	0	1	6	6	**24**	**6**	**163,333**	1	1	1	**3**	Low
						IR5.IT4.R5	2	0	2	6	12	**18**	**6**	**4,788**	1	2	2	**5**	Medium
						IR5.IT4.R6	3	1	4	6	24	**6**	**6**	**5,788**	1	2	2	**5**	Medium
						IR5.IT4.R7	1	0	1	6	6	**24**	**6**	**4,788**	1	2	2	**5**	Medium
DR1	2	0	2	6	12	DR1.DT1.R1	1	0	1	0	0	**12**	**6**	**47,880**	1	1	1	**3**	Low
						DR1.DT1.R2	0	0	0	0	0	**12**	**6**	**28,728**	1	2	1	**4**	Medium

5.5.3 Risk treatment-related concepts assessment

The assessment of security requirements can be done using two metrics - cost and risk reduction level. Besides, as the primary purpose of the security requirement is to treat the risks which affect business assets, security requirements can be characterised by the value of the business asset it aims to protect.

Estimate risk reduction level. The risk reduction level *RRL* characterises the impact of the security requirement on the risk level. To determine RRL, all the metrics of risk components should be estimated one more time but assuming the security requirement is met. The aim of reassessing the metrics is to analyse how the implementation of the requirement affects the risk components – the vulnerability level, the resource and know-how knowledge required to exert a threat, whether the requirement implementation reduces the impact on assets. Thereby, we perform the security risk assessment and calculate the risk level *RL* after the treatment. The risk reduction level is the difference between the initially assessed risk level and after security requirement implementation. The value of risk level equal to zero after the treatment means one of the following: (i) the addressed in the risk vulnerability is eliminated and likelihood to exert the threat is extremely low; (ii) the countermeasure eliminates impact on the targeted initially assets.

Estimate security requirement cost. The cost of security requirements primarily depends on the cost of control that implements the requirement. Additionally, the cost should include expenses on assessment and planning the requirement implementation (e.g., agreeing on the requirement with stakeholders, development tasks preparation). However, while the latter two sources of the cost may vary from one organisation to another depending on the followed system management and development procedures, the cost of the security controls are relatively fixed. Therefore, the cost of security requirements is estimated only based on the controls cost: $Cost = AVG(Cost_i)$, where $Cost_i$ is the cost of security control i which can be used to implement the requirement. The cost of controls is estimated by a security expert either based on the price of controls from the official websites of the service providers (e.g., RT1.C5.D.Log management system[2], ST1.C2.P. Biometric authentication for users[3]) or based on the complexity of countermeasure (how many months it can take to implement the security control) and the average salary in the EU[4].

5.5.4 Risk reduction strategy

As discussed in Sec. 5.4, we have elicitation in 52 security requirements that aim to reduce the risk level. From the management point of view, not all risks should be treated equally as the business has limited time and cost resources. One possible way to select which security requirements are first to be implemented is to prioritise them.

The requirements prioritisation should be fast and accurate in order to be applicable and useful in the system development context. Therefore, we compare the metrics of security requirements, related risk and delivered value, namely countermeasures *costs*, *risk reduction level* and the business asset *value*. These metrics allow us to assess the requirements based on their cost, quality (measured by the risk reduction level) and value to the business (measures through the value of protected assets).

Table 3 contains examples of the metrics required for the security requirements prioritisation, while the detailed calculations and the complete set of prioritised countermeasures can be found in [36]. Having the metrics values gathered, three graphs are created to compare RRL and cost, RRL and business asses

[2] `https://newrelic.com/pricing`. Accessed 30 Nov 2021

[3] `https://duo.com/editions-and-pricing`. Accessed 30 Nov 2021

[4] `https://ec.europa.eu/eurostat/databrowser/view/lc_lci_lev/` Accessed 30 Nov 2021

value, value and cost. Each graph is separated into four quadrants that separate low, medium, and high priority countermeasures. The security analyst selects the threshold values that separate requirements considering the limitations to expenses, time, and human resources when implementing the security features in the system.

Table 4 presents results of requirements separation when comparing the risk reduction level against the cost. The higher priority countermeasures have a high RRL for business assets with high value. The countermeasures which have high RRL but to the low-value assets or low RRL for high-value assets are classified as medium priority. Finally, countermeasures that deliver low RRL for low-value assets are classified as of low priority. For example, the requirements RT1.R1 and RT1.R2 are highly prioritised to reduce the risk RR1 (System Log Files Manipulation).

Table 4: Risk-reduction level against business asset value

		Risk Reduction Level (RRL)	
		$<$ 26	**$\geq$ 26**
Value	**$>$ 7**	***Medium priority*** R1.ST1.R3, TR6.TT3.R4, SR1.ST1.R4-7, RR1.RT1.R3-6; SR1.ST1.R1-2	***High priority*** RR1.RT1.R1-2; TR6.TT3.R1-3
	$\leq$ 7	***Low*** TR5.TT3.R4; TR4.TT2.R1-3, IR4.IT3.R1-4, IR5.IT4.R1, IR5.IT4.R4-7, ER4.ET4.R1-2; DR3.DT2.R1, IR1.IT1.R2, IR2.IT1.R2, TR3.TT1.R1-4, DR2.DT1.R1-2, DR4.DT3.R1-2, TR2.TT1.R1-4, DR1.DT1.R1-2; ER2.ET2.R1; TR1.TT1.R1-4, ER1.ET1.R1-3, ER1.ET1.R2.2, ER3.ET3.R1-4; DR5.DT4.R1	***Medium*** IR5.IT4.R1-3; IR3.IT2.R1-2; IR1.IT1.R1, IR2.IT1.R1, TR5.TT3.R1-3

In the table 5 the risk reduction level of countermeasures, the related costs are compared. The higher priority countermeasures have a high risk reduction level with low relative cost on its implementation. The countermeasures with high RRL and high cost or low RRL and low cost are classified as medium priority. In contrast, the countermeasure with high cost but low risk reduction levels are of low-priority solutions.

Table 5: Risk-reduction level against cost of countermeasure

		Risk Reduction Level (RRL)	
		$<$ 31	**$\geq$ 31**
Cost	**$>$ 30,000**	***Low priority*** SR1.ST1.R1, SR1.ST1.R4, RR1.RT1.R5, IR5.IT4.R4, DR1.DT1.R1, DR2.DT1.R1, ER1.ET1.R2.2	***Medium priority*** RR1.RT1.
	$\leq$ 30,000	***Medium priority*** SR1.ST1.R2, SR1.ST1.R3, SR1.ST1.R5, SR1.ST1.R6, SR1.ST1.R7, TR1.TT1.R1-4, TR2.TT1.R1-4, TR3.TT1.R1-4, TR4.TT2.R1-3, TR5.TT3.R4, TR6.TT3.R4, RR1.RT1.R3, RR1.RT1.R4, RR1.RT1.R6, IR1.IT1.R2, IR2.IT1.R2, IR4.IT3.R1-4, IR5.IT4.R1-3, IR5.IT4.R5-7, DR1.DT1.R2, DR2.DT1.R2, DR3.DT2.R1, DR4.DT3.R1-2, DR5.DT4.R1, ER1.ET1.R1-2, ER1.ET1.R3, ER2.ET2.R1, ER3.ET3.R1-4, ER4.ET4.R1-2	***High priority*** TR5.TT3.R1-3, TR6.TT3.R1-3, RR1.RT1.R1, IR1.IT1.R1, IR2.IT1.R1, IR3.IT2.R1-2

The comparison of protected assets' value and related costs on countermeasures are shown in Table 6.

The high priority requirements are to protect high-value assets with low related costs. Countermeasures that protect high-value assets with high related costs and low-value assets with low related costs are medium priority. Meantime, countermeasures protecting low-value assets but requiring high cost are of low priority.

Table 6: Risk-reduction level against cost of countermeasure

		Cost	
		$<$ 25,000	**$\geq$ 25,000**
Value	**$>$ 7**	***High priority*** SR1.ST1.R2-3, SR1.ST1.R5-7, TR6.TT3.R1-4, RR1.RT1.R12, RR1.RT1.R3-4, RR1.RT1.R6	***Medium priority*** RR1.RT1.R5, RR1.RT1.R2, SR1.ST1.R1
	$\leq$ 7	***Medium priority*** SR1.ST1.R4, TR1.TT1.R1-4, TR2.TT1.R1-4, TR3.TT1.R1-4, TR4.TT2.R1-3, TR5.TT3.R1-4, IR1.IT1.R1-2, IR2.IT1.R1-2, IR3.IT2.R1-2, IR4.IT3.R1-4, IR5.IT4.R1-3, IR5.IT4.R5-7, DR3.DT2.R1, DR4.DT3.R1-2, DR5.DT4.R1, ER1.ET1.R1-2, ER1.ET1.R3, ER2.ET2.R1, ER3.ET3.R1-4, ER4.ET4.R1-3	***Low priority*** IR5.IT4.R4, DR1.DT1.R1-2, DR2.DT1.R1-2, ER1.ET1.R2.2

Having defined priorities from the comparisons of cost, value and RRL, we convert priority into scores using a scale from 1 to 3 (low to high priority). The final priority score can be identified based on the sum of three previous priorities. The final score is depicted in Table 3 in the last column.

As a result, we have a set of prioritised security requirements using which a system owner can select a set to implement. We argue that requirements of medium and high priorities should be considered only to be implemented because they have better quality and a more positive impact on system security.

As we have assessed requirements' priority by the comparison of all the requirements, for some security risks, there can be few requirements of high priority. One should interpret it as that even though the risk has a relatively high-risk level, it can be treated with reasonable expenses. Still, we recommend picking only one high priority requirement per risk and implementing it in the system. Another thing to consider during the strategy selection is the risk profile and resource capabilities. If there are two countermeasures of high priority - one has higher RRL another has a higher cost - then the low-risk profile companies should prefer implementation of requirement with higher RRL unconditionally, while if there are expenses limitations, then the cheaper requirement should be preferred.

For instance, as presented in Table 3, the risk TR6 has four high prioritised requirements. Facing such a situation, a decision-maker who selects the risk reduction strategy should pick one of the requirements. Assuming the company prefers to reduce risks as much as possible, a decision-maker should select the requirements that enable the higher RRL (in case of the risk TR6 - TT3.R1 and TT3.R2) and pick one with a lower cost. Thereby, we propose the following strategy - to implement the requirement TT3.R1 to reduce the risk TR6.

To protect assets in the Passenger-AV interaction scenario, we argue that pursuing the following risk reduction strategy is beneficial. First, the company should treat the risks TR6 and RR1 by implementing TT3.R1 and RT1.R1, respectively. During the next security features development planning, the countermeasures also prioritised as of high priority but with a lower score can be considered. Thereby, high priority requirements are selected. At the next stage, after finishing with the most crucial risks' treatment, the decision-make may propose implementing the requirements of medium priority. Fig. 9 presents the risk reduction strategy as a product map highlighting the order in which security features should be integrated into the existing system.

Thereby, we have finished analysing the passenger-AV interaction scenario, which takes place during

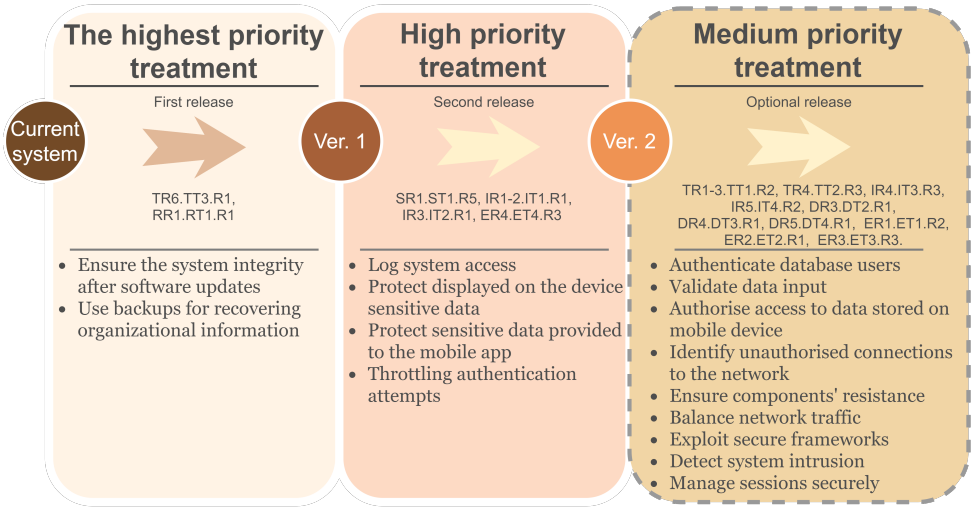

Figure 9: The map of security reduction strategy implementation

the Ride Fulfilment process in a ride-hailing company. The security risk management analysis results in the proposed security risks reduction strategy. This strategy includes system security requirements that should be implemented into the information systems used by the ride-hailing company and installed in the autonomous vehicle.

6 Conclusion

The study's primary goal is to determine how the information in the passenger–autonomous vehicle interaction can be protected. To address security risk management, we use a threat-driven approach to elicitation security requirements. As a result, in the previous work [4], we have developed a threat model for the passenger-AV interaction scenario that is supported by the information systems of a ride-hailing company. The previous work has resulted in the elicited set of security requirements based on the threat model. The security risk management activities are oriented on securing information in the AV's application layer and are limited to the Ride Fulfilment process where passenger directly interacts with an autonomous vehicle. However, similar security risks analysis should also be conducted in the other process (e.g., vehicle-to-infrastructure and vehicle-to-vehicle interactions) to assure the security of the moving AV.

The selection of a risk reduction strategy is an essential step of security risks management. As highlighted in [35], the development of advanced driving systems requires a systematic approach of security-aware safe systems. While security risks in the Ride Fulfilment process may affect the execution of the driving tasks, selecting an effective risk reduction strategy contributes to safety on the roads. The paper proposes the risk reduction strategy for securing the Passenger-AV interaction. We argue that the strategy should result from security risks and requirements assessment activity. The quality of security requirements can be estimated based on the risk reduction level. Even though some security requirements may allow us to reduce the risk or even avoid it, such requirements may be expensive or aim to protect assets of low value. Therefore, besides the impact on risks, the cost of implementation and protected asset value should be considered during the prioritisation. Moreover, the results of security risks assessment can be used as an additional source for safety risks assessment.

Threats to validity. The research is conducted using the case study analysis that allows us to investigate autonomous driving usage. This method is applicable as the field is relatively new, and the

technology is still developing. Consequently, the phenomenon is not yet determined, and the approaches for tackling it are not standardised either. Although, the results are case-oriented and require validation on the other cases of the AV by passengers that threaten the external validity of the paper results.

Another limitation of the study results is the factor of subjectivity during the metrics assessment. Risk assessment results rely on experts' opinions to estimate such parameters as assets' impact areas levels, vulnerability level, level of required resources and knowledge for threat implementation. The costs of security controls are calculated based on the average labour cost in the European Union and the prices of solutions available on service providers' websites that may vary from country to country and could be negotiable. The interpretation of countermeasures impact may be threatened with the researcher's subjective assessment.

Future Work. In this study, we considered the complex heterogeneous system consisting of an AV system, an external service provider (a ride-hailing company in our case), and a human (passenger). Consequently, treatment of security risks should be applied not only towards the information systems. The risk treatment should also include extensions to the organisational security policies and user guidelines. Having defined the risk reduction strategy as a prioritised set of system security requirements, we will investigate how the strategy should be extended with the organisational measures in future work. Additionally, the proposed strategy will be checked into compliance with the newly released standard in the automotive industry – ISO/SAE 21434. Finally, we aim to develop a framework for assuring security-aware autonomous vehicle usage by the service providers to help business companies integrate AVs into their system infrastructure and business processes.

Acknowledgments

This paper has been supported in part by EU Horizon 2020 research and innovation programme under grant agreement No 830892, project SPARTA, and European Social Fund via "ICT programme" measure.

References

[1] M.A. Assaad, R. Talj, and A. Charara. Autonomous driving as system of systems: roadmap for accelerating development. In *Proc. of the 14th Annual Conference System of Systems Engineering (SoSE'19), Anchorage, Alaska, USA*, pages 102–107. IEEE, May 2019.

[2] On-Road Automated Driving (ORAD) committee. Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles. Technical Report J3016, SAE international, 2021.

[3] ENISA. Enisa good practices for security of smart cars. Technical report, European Union Agency for Cybersecurity, 2019.

[4] M. Bakhtina and R. Matulevičius. Information security analysis in the passenger-autonomous vehicle interaction. In *Proc. of the 16th International Conference on Availability, Reliability and Security (ARES'21), Vienna, Austria*, pages 1–10. ACM, August 2021.

[5] ISO. Iso/sae 21434:2021 road vehicles — cybersecurity engineering, August 2021. `{https://www.iso.org/standard/70918.html}[Online;accessedonMarch15,2022]`.

[6] T. Soiunen. Design and evaluation of a user interface to increase trust in autonomous vehicles. Master's thesis, University of Tartu, May 2020.

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

[8] W. Abbass, A. Baina, and M. Bellafkih. Survey on information system security risk management alignment. In *Proc. of the 2nd International Conference on Information Technology for Organizations Development (IT4OD'16), Fez, Morocco*, pages 1–6. IEEE, March-April 2016.

[9] E. Dubois, P. Heymans, N. Mayer, and R. Matulevičius. A systematic approach to define the domain of information system security risk management. In *Intentional Perspectives on Information Systems Engineering*, pages 289–306. Springer, Berlin, Heidelberg, May 2010.

[10] D. Ganji, H. Mouratidis, and S.M. Gheytassi. Towards a modelling language for managing the requirements of ISO/IEC 27001 standard. In *Proc. of the 5th International Conference on Advances and Trends in Software Engineering (SOFTENG'19), Valencia, Spain*, pages 17–23. IARIA, May 2019.

[11] V.L.L. Thing and J. Wu. Autonomous vehicle security: A taxonomy of attacks and defences. In *Proc. of the 7th IEEE International Conference on Internet of Things (iThings'16) and IEEE Green Computing and Communications (GreenCom'16) and IEEE Cyber, Physical and Social Computing (CPSCom'16) and IEEE Smart Data (SmartData'16), Chengdu, China*, pages 164–170. IEEE, December 2016.

[12] Z. El-Rewini, K. Sadatsharan, D.F. Selvaraj, S.J. Plathottam, and P. Ranganathan. Cybersecurity challenges in vehicular communications. *Vehicular Communications*, 23:100214–100241, June 2020.

[13] J. Petit and S.E. Shladover. Potential cyberattacks on automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 16(2):546–556, April 2015.

[14] S.Parkinson, P. Ward, K.M. Wilson, and J. Miller. Cyber threats facing autonomous and connected vehicles: Future challenges. *IEEE Transactions on Intelligent Transportation Systems*, 18(11):2898–2915, November 2017.

[15] S.K. Khan, N. Shiwakoti, P. Stasinopoulos, and Y. Chen. Cyber-attacks in the next-generation cars, mitigation techniques, anticipated readiness and future directions. *Accident Analysis & Prevention*, 148:105837–105852, December 2020.

[16] IPA. Approaches for vehicle information security. Technical report, Information technology-Promotion Agency, 2013.

[17] C. Hodge, K. Hauck, S. Gupta, and J.C. Bennett. Vehicle cybersecurity threats and mitigation approaches. Technical Report NREL/TP-5400-74247, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2019.

[18] ENISA. Recommendations for the security of cam. Technical report, European Union Agency for Cybersecurity, 2021.

[19] M. Bakhtina. Securing passenger's data in autonomous vehicles. Master's thesis, University of Tartu, June 2021.

[20] A. Shostack. *Threat Modeling: Designing for Security*. Wiley Publishing, 2014.

[21] M.E. Whitman and H.J. Mattord. *Principles of Information Security*. Cengage Learning, 2012.

[22] MITRE. Capec - common attack pattern enumeration and classification, March 2022. `{https://capec.mitre.org/}[Online;accessedonMarch15,2022]`.

[23] Microsoft. The STRIDE Threat Model, December 2009. `{https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20)}[Online;accessedonMarch15,2022]`.

[24] MITRE. Mitre att&ck, March 2022. `{https://attack.mitre.org/}[Online;accessedonMarch15,2022]`.

[25] OWASP. OWASP Top Ten, October 2021. `{https://owasp.org/www-project-top-ten/}[Online;accessedonMarch15,2022]`.

[26] O. Altuhhova, R. Matulevičius, and N. Ahmed. An extension of business process model and notation for security risk management. *International Journal of Information System Modeling and Design*, 4(4):93–113, October 2013.

[27] D. Firesmith. Engineering security requirements. *Journal of Object Technology*, 2(1):53–68, January 2003.

[28] National Highway Traffic Safety Administration. Cybersecurity best practices for modern vehicles (dot hs 812 333). Technical Report DOT HS 812 333, National Highway Traffic Safety Administration, 2016.

[29] CIS. Critical security controls, version 7.1, 2019. `{https://learn.cisecurity.org/cis-controls-download}[Online;accessedonMarch15,2022]`.

[30] Vehicle Cybersecurity Systems Engineering Committee. Cybersecurity guidebook for cyber-physical vehicle systems (j3061). Technical Report 3061, SAE International, 2016.

[31] P.A. Grassi et al. Digital identity guidelines: authentication and lifecycle management. Technical report, National Institute of Standards and Technology, 2017.

[32] Joint Task Force. Security and privacy controls for information systems and organizations (sp 800-53). Technical Report SP 800-53, National Institute of Standards and Technology, 2020.

[33] A. Bassi and Others. Flying 2.0 enabling automated air travel by identifying and addressing the challenges of iot & rfid technology. Technical report, ENISA, 2010.

[34] MITRE. Cwe - common weakness enumeration, March 2022. {https://cwe.mitre.org/}[Online; accessedonMarch15,2022].

[35] G. Macher, H. Sporer, R. Berlach, E. Armengaud, and C. Kreiner. Sahara: a security-aware hazard and risk analysis method. In *Proc. of the 18th Design, Automation & Test in Europe (DATE'15), Grenoble, France*, pages 621–624. ACM, March 2015.

[36] M. Bakhtina. Calculations for security risk assessment and requirements prioritization (passenger-AV interaction scenario), February 2022. {https://doi.org/10.5281/zenodo.6077743}[Online; accessedonMarch15,2022].

Author Biography

Mariia Bakhtina received the M.A. degree in innovation and technology management from the University of Tartu (UT), Estonia. There, she is pursuing the Ph.D. degree in computer science. Also, she is working as a Junior Research Fellow with UT. Her research interests include the influence of technologies and digital products on organisations, particularly how intelligent systems should be managed in terms of information security and privacy.

Raimundas Matulevičius received the Ph.D. degree in computer and information science from the Norwegian University of Science and Technology. He currently holds a Professor of information security position at the University of Tartu, Estonia. His publication record includes more than 100 articles published in peer-reviewed journals, conferences, and workshops. He is the author of a book Fundamentals of Secure System Modelling (Springer, 2017). He is involved in the SPARTA H2020 Project, ERASMUS+ Sectoral Alliance Program CHAISE, and Erasmus+ Strategic Partnership program CyberPhish. His research interests include security and privacy of information, security risk management, security and privacy by design and model-driven security. He has been a Program Committee Member at international conferences (e.g., CAiSE, NordSec, PoEM, and other) and a member of the editorial boards of recognised journals (e.g., REEN and BISE both by Springer).