Safety & Security Alignment in Requirements Engineering Process for Autonomous Vehicles

Quelita Ribeiro[1[0000-0002-5045-3660]] and Jaelson Castro[1[0000-0002-4635-7297]]

[1] Universidade Federal de Pernambuco, Recife, PE - Brazil
{qadsr, jbc}@cin.ufpe.br

Abstract. Autonomous Vehicles (AVs) will transform the way we live and work. In order to cope with the complexity of AVs, we want to pro- vide an understandable process for organizations to analyze the areas of safety and security in requirements engineering for AVs. Initially, a bibliographic survey was carried out to verify the current state of the art. Based on this survey, we carried out a systematic literature review to find answers to our questions and better understand the requirements engineering context for AVs. From the review responses, a process will be built to unite safety and security analysis for AV. STPA, and Misuse Case techniques will be verified, and the advantages can be used in our process. Finally, we intend to evaluate and validate our re- search within the industry with requirements engineers in the automotive sector. This work is a step towards developing a body of knowledge in RE for AVs.

Keywords: Autonomous Vehicle · Requirements Engineering · Safety · Security · STPA · Misuse Case.

Level: PhD
Year of program entry: 2020.1
Expected completion time: 2024.2

1 Introduction

Autonomous Vehicle (AV) is a car that makes driving decisions without the intervention of a human. As such, autonomy exists in many aspects of a car today: cruise control and anti-lock brake systems are excellent examples of systems that exhibit autonomous behavior. Additional systems already exist on some models, including advanced cruise control, lane maintenance support, lane change warning and obstacle prevention systems, all of which expand the range of autonomous behavior [22].

The AV consists of orchestration of hardware components with complex software implementations and multiple internal networks connecting intelligent sensor nodes with electronic control units and actuators. Therefore, many messages are exchanged within milliseconds [25]. Such connectivity and cooperation enable vehicles to connect with other road users and the infrastructure systems [27]. Furthermore, critical data dependencies exist, e.g., components that are mandatory for the core functionali-

ty of the vehicle communicate with noncritical components that provide comfort features for the passengers [25].

According to [31] automated driving will change the future's private transport and is currently the most discussed and disruptive technology in the automotive domain. Indeed, AVs are attracting more interest with each passing day in the industry, as well as the public. For [8], a system of driverless vehicles will change the global automotive sector where autonomous vehicle development will be a continuously evolving domain that will drastically alter the way people and goods are transported, ownership of vehicles is acquired and how ride-sharing services are utilized in the future.

Given the long development time for an AV, it may take a considerable time before a prototype becomes available to experiment with. Hence, it is important to get the requirements right from the beginning because it contributes to reduce costs, resources, time, and effort in the other phases of software development, mainly in the software development and testing. However, very few studies have focused on expressing the requirements of AV at a high level. Moreover, system building is subject to requirements change according to the practice tests on the prototype. Therefore, a critical issue is how to consider those changes and integrate them in the current specification [29].

1.1 Motivations and Rationale

It is well known that inappropriate RE practices may result in incomplete requirements, incorrect elicitation and specification, high complexity, and economic or human loss. Hence, it is necessary to investigate how RE is being adapted to deal Autonomous Vehicles to avoid inadequate or misunderstood requirements. RE for AV is challenging since it has unique properties that make it complex, expensive and error prone as compared with other categories, such as information systems [16], [23].

The rising complexity of automated driving functions makes it hard to define all concerning requirements in detail, formulate the development goal in more than an abstract way, as well as to estimate the development effort [31]. Therefore, requirements engineering problems in the domain of AVs continue to occur despite the efforts and advances in their understanding. Due to their properties, different approaches, methods, and tools are required to improve their quality. Some studies provide insights into the practice of RE for AVs [11], [31], [10], [28]. Autonomous vehicles differ from other systems in that they have several main concerns, and their operation is dynamic, for example they need to comply with federal and state laws in their region. In case the car leaves your region, any necessary changes must be determined and integrated into the system as safety requirements, including relevant speed limits, traffic signs and other signs. There- fore, it is necessary to improve security analysis techniques, ethics, certification, transparency in the process due to a collision or other situation, traceability for specifying requirements and fast retrieval of information about the requirements.

Furthermore, the autonomous vehicle is a novelty with several challenges and doubts.

The complexity of AVs, their connection to external networks, to the internet of things, and internal network opens doors to hackers and malicious attacks. Therefore, violation of security could lead to safety violations [12]. There are works that try to define a security engineering process along the lines of a safety engineering process for automotive vehicle systems [3], [6], [9], [12]. The idea is that the processes are similar and that they could be laid side-by-side and could be performed together - but, by a different set of experts [12].

Hence, industry challenges in the ER process of autonomous vehicles have motivated research related to the improvement of the requirements engineering process, regarding the safety and security of such systems.

1.2 Objectives

Based on the context and motivations presented, the main research question proposed to be investigated by this thesis is:

Research Question: How to provide a Requirements Engineering Process (called **RE4AV**) to be used in organizations that develop Autonomous Vehicles, focused on Safety and Security?

In order to answer this Research Question, the following specific objectives are proposed:

- O1 Perform comprehensive analysis of important authors in the field and conduct SLR;
- O2 Analyze safety standards;
- O3 Analyze security standards;
- O4 Study the current alignment processes between safety & security for autonomous vehicles;
- 05 Consider the use of STPA approach and Misuse Case as a means to align safety & security requirements;
- 06 – Define the **RE4AV** Requirements Engineering Process, including the steps/activities to be performed, as well as the generated artifacts (consider using BPMN to describe this process);
- 07 Develop a tool to support the proposed **RE4AV** Proces;
- 08 Validate and evaluate the proposed **RE4AV** Process.

2 Basic concepts

Autonomous vehicles are also known as self-adaptive software systems, automated driving system and belongs to the cyber physical system family.

In the past few years, several papers have been presented to support the development of AVs, such as models [34], [15], languages [21], [26], and tools [2]. Systems

composed by many autonomous components, that are called to operate in a coordinated way in open and unpredictable environments [4], [1].

The development of AV requires interdisciplinary cooperation between different stakeholders. A lack of system understanding between stakeholders can lead to unidentified security threats & safety hazards in requirements engineering, resulting in high costs in product development. A lack of an integrative consideration of security threats & safety hazards can compromise safety compliance for AV [16].

For example, in 2015 hackers demonstrated an attack on a moving SUV [14], [16]. In this case, the infotainment system was compromised by a remote hack, allowing the hackers to take control of the vehicle. This triggered a product recall of 1.4 million vehicles for the affected company [13], [16].

Governing bodies and standards organizations create regulations and standards to address issues such as safety, security, and privacy. In this environment, the compliance of software development to standards and regulations has emerged as a key requirement [7]. Nevertheless, no standard provides a structured co-engineering process to facilitate the communication between safety and security engineers. Since vehicles provide highly interconnected system functions realized in software, the systems are no longer isolated [5].

Development of AV begins with hazard analysis, aimed to identify possible causes of harm. It uses severity, probability, and controllability of a hazard's occurrence to assign the Safety Integrity Levels (these are referred to as ASILs [32])

– the higher the more rigor is expected to be put into identifying and mitigating the hazard [7]. Mitigating hazards therefore becomes the main requirement of the system, with system safety requirements being directly linked to the hazards [7]. Whereas safety deals with hazards and mishaps cybersecurity addresses threats resulting from malicious intent from external to the E/E system [5].

Safety discipline considers systematic and random hardware failures as hazard sources. Security considers a malicious and intelligent adversary as a threat source in addition to natural disasters and systematic failures. The unacceptable consequences for safety are loss of human life and injuries. Security as a discipline has a broader range of unacceptable consequences: human life, human security, loss of reputation, financial losses, legal violations, loss of intellectual property, damage critical infrastructure etc. These differences could be reconciled [12].

However, established security & safety approaches are either only applicable to specific disciplines or only partially consider security threats & safety hazards [16]. Furthermore, the reuse of security & safety solution knowledge to reduce the engineering effort is not considered in literature [16].

To [25], it is necessary an integrated evaluation of safety and security on a designated abstraction level. Due to the complexity of the system and its hard safety requirements. Moreover, safety and security are partly intertwined. A safety failure of a component may favor the successful attack on another component, e.g., if the Hardware Security Module (HSM) fails and security mechanisms such as authentication cannot be provided anymore, the risk for a successful attack rises. This interdependence of safety and security especially in the automotive domain has already been widely recognized [12], [6], [5], [25].

2.1 Safety and Security Techniques

System-Theoretic Process Analysis (STPA) is a technique proposed by [19] to analyze safety. STPA considers that accidents can be caused by the interaction between system components, even without failures. In this sense, in STPA, it is necessary to control the behavior of the individual components and their interactions. There is then a controller to enable the evaluation of control actions of the components and the respective feedbacks in relation to the safety of the system as a whole. The controller will enforce restrictions on system behavior. [20]. Figure 1 shows the idea of a standard STPA controller.

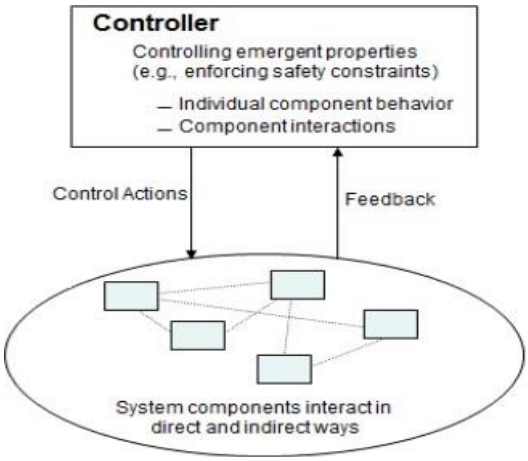

Figure 1: Standard STPA controller - Fonte: [20].

Use cases (with slight modifications) can aid the integration of functional and non-functional requirements work when considering security requirements. Misuse cases specify behavior not wanted in the proposed system for the purpose of eliciting security requirements [30]. Many security breaches can be described in a stepwise fashion in an unwanted interaction sequence [30]. For example, the use case is a countermeasure against a misuse case, i.e., the use case reduces the misuse case's chance of succeeding. An example is "protect info", which mitigates "steal credit card info", as shown in Fig. 2.

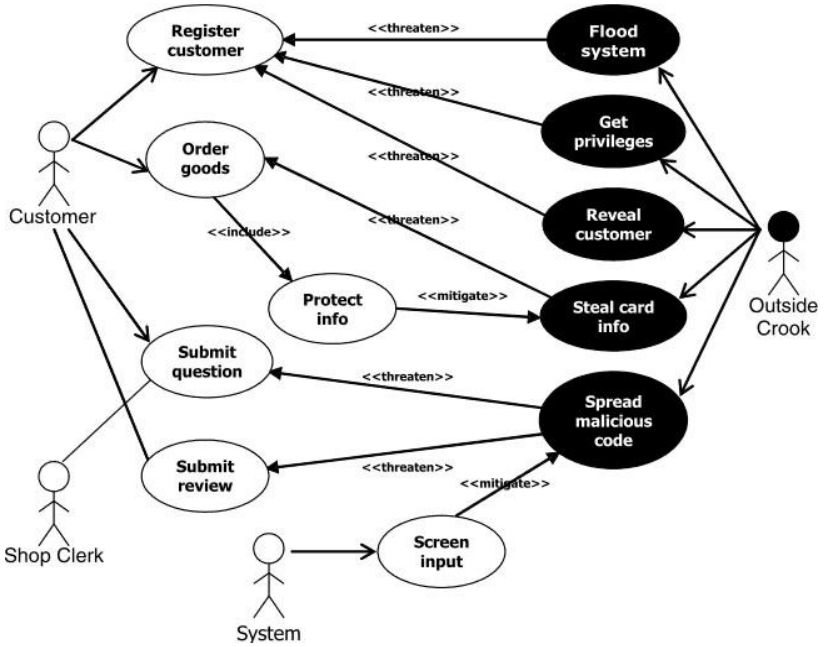

Figure 2: Example use and misuse cases for an e-store - Fonte: [30].

3 Research Methodology

In this work, we use some research method techniques. The first one was the literature review. According to [33], the literature review presents the concepts necessary for understanding the objective and the works related to the objective. For this reason, we started searches for periodicals, technical articles, journals, and theses.

After that, we prepared the systematic review protocol and carried out the Systematic Literature Review (SLR) to discover the answers to the questions proposed in this work.

In order to perform this SLR, we used guidelines and the protocol template proposed by Kitchenham, and Charters [17], whose process involves several activities grouped into three main phases: planning, conducting, and reporting of the review. It consists of the following steps: (1) identification of the need for a systematic review, (2) development of a review protocol, (3) a comprehensive, exhaustive search for primary studies, (4) quality assessment of included studies, (5) data extraction and monitoring, (6) data analysis and synthesis, and (7) report-writing.

Therefore, we synthesized our research in a SLR. To [18], the main method of synthesis is a SLR. In contrast to an expert review using ad hoc literature selection, an SLR is a methodologically rigorous review of research results.

SLR is part of the Evidence-Based Software Engineering (EBSE) research method. Evidence-based research and practice was developed initially in medicine because

research indicated that expert opinion based medical advice was not as reliable as advice based on the accumulation of results from scientific experiments [18].

Goal of EBSE is to provide the means by which current best evidence from research can be integrated with practical experience and human values in the decision-making process regarding the development and maintenance of software. The end point of EBSE is for practitioners to use the guidelines to provide appropriate software engineering solutions in a specific context [18].

Our SLR aims to identify and analyze the current RE approaches for AVs. The analysis is based on RQs answers related to the type of RE problems addressed by the study, the RE phases covered by the approach, requirements modelling styles used, the type of requirements described in the study, the specific AVs considered in the RE study, and the open problems reported.

An automatic search was conducted in the following electronic databases: ACM Digital Library, IEEE Xplore, Science Direct, Scopus, and Springer.

We collect information about requirements engineering for AVs. Thus, we had focused on terms of the RE area, autonomous vehicles, and kind of contribution. With the answers to the questions and the synthesis of the SLR in hand, we are starting to create our requirements engineering process for autonomous vehicles with the achievement of safety and security that encompasses the activities included in the RE phases for AV. Requirements description style is still being discussed, but we want to use a language that is easy for stakeholders to understand.

Finishing our process, we will make use of the survey research technique, which consists of searching for existing data directly in the environment, through observations, measurements, questionnaires, and interviews [33]. Thus, after tabulating this information, conclusions can be drawn about causes and effects [33]. Thus, we will evaluate the methodology in the industry to evaluate the approach created. Furthermore, we want to obtain and analyze the evaluation results to promote improvements to the proposed method and receive feedback from engineers.

4 Current Status of Work

The first phase of the bibliographic survey work with knowledge acquisition and the second phase, which includes the systematic literature review, were completed (See [24]).

The bibliographic survey and the SLR allowed the visualization of the state of the art and the discovery of the problems and challenges faced by the AV industry, the gap between safety and security, the need to align these two non- functional requirements and, finally, the need an easy language for AV engineering teams to communicate.

We are currently working on developing the safety & security achievement process. We are still studying some safety and security techniques separately. STPA (safety) and Misuse Cases (security) have been widely used in the literature. In this way, we are verifying the possibility of uniting the advantages of the two techniques and proposing something in this sense.

5 Expected Contributions

The main objective of this work is to provide a **RE4AV** process for organizations to conduct Requirements Engineering focused on safety and security of AVs. Moreover, we want to develop a tool to support the application of the safety & security process. Finally, validate and evaluate the **RE4AV** process.

We hope that our process will be used by requirements engineers in the automotive industry to align safety & security analysis in the early stage of AV development, which will increase the trust and success of stakeholders and customers.

6 Comparison with Related Works

In [16], Model-Based Requirements Engineering (MBRE) is considered to improve the understanding of systems among stakeholders by creating models to support system requirements. However, MBRE approaches only partially ad- dress security threats & safety hazards. Besides that, integrative consideration between safety & security is not considered.

It is presented in [25], a combination of an analytical approach using Markov chains with a numerical approach using simplified state diagrams to model the system structure and a Monte Carlo simulation to analyze the system's behavior. The goal was to develop a method that is convenient for performing a holistic safety and security evaluation of autonomous vehicles with a simple example. In our work, we are concerned with the development of activities for all phases of requirements engineering for AVs, also considering safety & security analysis. In addition, we are focused on developing a process with an easy language for stakeholders to communicate.

7 Acknowledgements

The authors are grateful for the financial support of Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES), Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) and Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco (FACEPE).

References

1. Abeywickrama, D.B., Bicocchi, N., Mamei, M., Zambonelli, F.: The sota approach to engineering collective adaptive systems. International Journal on Software Tools for Technology Transfer 22(4), 399-415 (2020)
2. Abeywickrama, D.B., Hoch, N., Zambonelli, F.: Engineering and implementing software architectural patterns based on feedback loops. Scalable Computing: Practice and Experience 15(4), 291-308 (2014)

3. Avizienis, A., Laprie, J.C., Randell, B., Landwehr, C.: Basic concepts and taxon omy of dependable and secure computing. IEEE transactions on dependable and secure computing 1(1), 11-33 (2004)
4. Belzner, L., H¬®olzl, M., Koch, N., Wirsing, M.: Collective autonomic systems: Towards engineering principles and their foundations. In: Transactions on Foundations for Mastering Change I, pp. 180-200. Springer (2016)
5. Bramberger, R., Martin, H., Gallina, B., Schmittner, C.: Co-engineering of safety and security life cycles for engineering of automotive systems. ACM SIGAda Ada Letters 39(2), 41-48 (2020)
6. Burton, S., Likkei, J., Vembar, P., Wolf, M.: Automotive functional safety= safety+ security. In: Proceedings of the First International Conference on Security of Internet of Things. pp. 150-159 (2012)
7. Chechik, M., Salay, R., Viger, T., Kokaly, S., Rahimi, M.: Software assurance in an uncertain world. In: International Conference on Fundamental Approaches to Software Engineering. pp. 3-21. Springer, Cham (2019)
8. Cysneiros, L.M., Raffi, M., do Prado Leite, J.C.S.: Software transparency as a key requirement for self-driving cars. In: 26th International Requirements Engineering Conference (RE). pp. 382-387. IEEE (2018)
9. Czerny, B.J.: System security and system safety engineering: Differences and sim ilarities and a system security engineering process based on the iso 26262 process framework. SAE International Journal of Passenger Cars-Electronic and Electrical Systems 6(2013-01-1419), 349-359 (2013)
10. Daun, M., Stenkova, V., Krajinski, L., Brings, J., Bandyszak, T., Weyer, T.: Goal modeling for collaborative groups of cyber-physical systems with grl: reflections on applicability and limitations based on two studies conducted in industry. In: Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing. pp. 1600-1609 (2019)
11. Emmerich, O., Wang, H., Garcia, G., Pezzulla, I., Darlington, A., Gao, B.: A systems engineering framework and application to an open automated driving platform. In: IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society. pp. 2393-2398. IEEE (2020)
12. Glas, B., Gebauer, C., H¬®anger, J., Heyl, A., Klarmann, J., Kriso, S., Vembar, P., Worz, P.: Automotive safety and security integration challenges. Automotive- Safety & Security 2014 (2015)
13. Goldman, D.: Chrysler recalls 1.4 million hackable cars. 2015, CNN Business, URL https://money.cnn.com/2015/07/24/technology/chrysler-hack-recall/index.html (2015), access: 26.04.2022.
14. Greenberg, A.: Hackers remotely kill a jeep on the highway-with me in it (2015). URL https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway (2019), access: 26.04.2022.
15. Herbst, N., Becker, S., Kounev, S., Koziolek, H., Maggio, M., Milenkoski, A., Smirni, E.: Metrics and benchmarks for self-aware computing systems. In: Self- Aware Computing Systems, pp. 437-464. Springer (2017)
16. Japs, S.: Security & safety by model-based requirements engineering. In: 2020 IEEE 28th International Requirements Engineering Conference (RE). pp. 422-427. IEEE (2020)
17. Kitchenham, B., Charters, S.: Guidelines for performing systematic literature re- views in software engineering (2007)
18. Kitchenham, B., Brereton, O.P., Budgen, D., Turner, M., Bailey, J., Linkman, S.: Systematic literature reviews in software engineering-a systematic literature review. Information and software technology 51(1), 7-15 (2009)

19. Leveson, N.G.: Engineering a Safer World: Systems Thinking Applied to Safety. Mit Press, Massachusetts, London, England (2011)
20. Leveson, N.G., Thomas, J.P.: STPA Handbook. first ed n. (2018)
21. Nicola, R.D., Loreti, M., Pugliese, R., Tiezzi, F.: A formal approach to autonomic systems programming: the scel language. ACM Transactions on Autonomous and Adaptive Systems (TAAS) 9(2), 1-29 (2014)
22. Ozguner, U., Stiller, C., Redmill, K.: Systems for safety and autonomous behavior in cars: The darpa grand challenge experience. Proceedings of the IEEE 95(2), 397-412 (2007)
23. Ramesh, S., Vogel-Heuser, B., Chang, W., Roy, D., Zhang, L., Chakraborty, S.: Specification, verification and design of evolving automotive software. In: Proceedings of the 54th Annual Design Automation Conference 2017. pp. 1-6 (2017)
24. RIBEIRO, Q.A., RIBEIRO, M., CASTRO, J.: Requirements engineering for autonomous vehicles: a systematic literature review. In: Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing. pp. 1299-1308. ACM (2022)
25. Rinaldo, R.C., Horeis, T.F.: A hybrid model for safety and security assessment of autonomous vehicles. In: Computer Science in Cars Symposium. pp. 1-10 (2020)
26. Salvaneschi, G., Ghezzi, C., Pradella, M.: Contexterlang: A language for distributed context-aware self-adaptive applications. Science of Computer Programming 102, 20-43 (2015)
27. Schmittner, C., Ma, Z., Schoitsch, E., Gruber, T.: A case study of fmvea and chassis as safety and security co-analysis method for automotive cyber-physical systems. In: Proceedings of the 1st ACM Workshop on Cyber-Physical System Security. pp. 69-80 (2015)
28. Semmak, F., Gnaho, C., Brunet, J., Laleau, R.: How to adapt the kaos method to the requirements engineering of cycab vehicle. In: ENASE. pp. 87-94 (2009)
29. Semmak, F., Gnaho, C., Laleau, R.: Extended kaos method to model variability in requirements. In: Evaluation of Novel Approaches to Software Engineering, pp. 193-205. Springer (2009)
30. Sindre, G., Opdahl, A.L.: Eliciting security requirements with misuse cases. Requirements engineering 10(1), 34-44 (2005)
31. Sippl, C., Bock, F., Lauer, C., Heinz, A., Neumayer, T., German, R.: Scenario- based systems engineering: an approach towards automated driving function development. In: 2019 IEEE International Systems Conference (SysCon). pp. 1-8. IEEE (2019)
32. Standard, I.: 26262 road vehicles-functional safety. International Organization for Standardization, www. iso. org, date of access 20171210 (2018)
33. Wazlawick, R.: Metodologia de pesquisa para ciÀÜencia da computa¬∏cÀúao, vol. 2. Elsevier Brasil (2017)
34. Weyns, D., Malek, S., Andersson, J.: Forms: Unifying reference model for formal specification of distributed self-adaptive systems. ACM Transactions on Autonomous and Adaptive Systems (TAAS) 7(1), 1-61 (2012)