In Search of Synergies in a Multi-Concern Development Lifecycle: Safety and Cybersecurity

Martin Skoglund, Fredrik Warg and Behrooz Sangchoolie

Department of Electronics, RISE Research Institutes of Sweden, Borås, Sweden
{martin.skoglund,fredrik.warg,behrooz.sangchoolie}@ri.se

Published in:

Proceedings of SAFECOMP 2018 Workshops, ASSURE, DECSoS, SASSUR, STRIVE, and WAISE, Västerås, Sweden, September 18, 2018, pp 302-313, Springer International Publishing.

The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-99229-7_26

In Search of Synergies in a Multi-Concern Development Lifecycle: Safety and Cybersecurity

Martin Skoglund[1[0000-1111-6901-4986]], Fredrik Warg[1[0000-0003-4069-6252]] and Behrooz Sangchoolie [1[0000-0001-9536-4269]]

[1] Department of Electronics, RISE Research Institutes of Sweden, Borås, Sweden
{martin.skoglund,fredrik.warg,behrooz.sangchoolie}@ri.se

Abstract. The complexity of developing embedded electronic systems has been increasing especially in the automotive domain due to recently added functional requirements concerning e.g., connectivity. The development of these systems becomes even more complex for products - such as connected automated driving systems – where several different quality attributes (such as functional safety and cybersecurity) need to also be taken into account. In these cases, there is often a need to adhere to several standards simultaneously, each addressing a unique quality attribute. In this paper, we analyze potential synergies when working with both a functional safety standard (ISO 26262) and a cybersecurity standard (first working draft of ISO/SAE 21434). The analysis is based on a use case developing a positioning component for the automotive domain. The results regarding the use of multi-concern development lifecycle is on a high level, since most of the insights into co-engineering presented in this paper is based on process modeling. The main findings of our analysis show that on the design-side of the development lifecycle, the big gain is completeness of the analysis when considering both attributes together, but the overlap in terms of shared activities is small. For the verification-side of the lifecycle, much of the work and infrastructure can be shared when showing fulfillment of the two standards ISO 26262 and ISO/SAE 21434.

Keywords: Functional Safety, Cybersecurity, Automotive, Co-engineering, Multi-concern.

1 Introduction

Synergies between different development processes for increasingly more complex, interconnected and intelligent cyber-physical systems is needed to increase quality attributes and reduce time to market. Examples are connected automated driving systems, robots and intelligent manufacturing technologies (Industry 4.0). The increasing number of highly automated systems bring many opportunities, e.g. interconnection of individual systems to create system-of-systems enabling new functionality, or reuse of components between products in different domains to bring down costs for introducing automation. However, such cyber-physical systems are often dependability-critical, i.e., care must be taken during development to make sure relevant quality

attributes such as safety, availability, robustness and cybersecurity are adequately met, to avoid unreasonable risk of harm to persons, the environment or economy.

In many cases, the means of meeting a quality attribute is developing a product according to a standard, and standards are typically tailored to be both domain and quality attribute specific. ISO 26262 [1] is a standard used for ensuring functional safety in the automotive domain. This means if several quality attributes, or reuse in several domains, is desirable, it may be necessary to show conformance to several standards for the same product. However, conforming to standards typically carries some overhead for product development and using several standards will exacerbate this problem. Therefore, the aim is to find *synergies* in the form of activities, techniques, or measures which simultaneously satisfies requirements in more than one of the standards used.

An example when using several standards for the same quality attribute may be a supplier wanting to sell the same system or component to original equipment manufacturers in different domains, making it necessary to show conformance to e.g. ISO 26262 when targeting the automotive domain (on-road) and IEC 61508 [2] for machinery (off-road). It has already been shown that safety standards for different domains have a significant overlap, making it possible to reuse the work done for one standard when targeting another [3]. In this paper, we investigate the possibility of finding overlaps between the work that needs to be done to address different quality attributes. The integrated development process for several quality attributes is in this paper denoted as *multi-concern development* and the act of providing confidence that the risks have been reduced to an acceptable level is denoted as *multi-concern assurance*. As the aim for each concern is different, it is expected to be challenging to find synergies when performing multi-concern development.

Due to the nature of cyber-physical systems that are connected and highly automated, the combination of safety and cybersecurity is of high interest. It is important to realize that quality attributes can also exhibit dependencies. For instance, a security problem may allow a hacker to disable or fool a safety mechanism. Therefore, both concerns as well as their *interplay*, must be properly addressed. We analyze the potential for synergies in a multi-concern development lifecycle. The analysis is based on a case study where a product for the automotive domain is developed according to both a safety standard (ISO 26262) and a security standard (first working draft of ISO/SAE 21434 [4]). The ISO/SAE 21434 is intended to address security for road vehicles, in similar way as the ISO 26262 standard does for functional safety. The development of an automotive domain specific cybersecurity standard is followed with close interest within the automotive industry and is assumed to have a high impact how security is handled. Therefore, the content of the standard, even at draft stage, becomes relevant and interesting for analysis.

For the design phases, i.e. the left side of the development V-model used in both standards, we found the overlap between safety and cybersecurity in terms of analysis, countermeasures and requirements to be small. For this part of the development lifecycle, the main advantage of treating both concerns in parallel is completeness, i.e. that both concerns and their interplay are considered through all stages, thereby reducing the risk of missing issues that may cause costly major redesigns if discovered

later, or even worse, resulting in too high residual risks if dependencies are not discovered at all. For the right side of the V model, i.e. the verification and validation side, the possible synergies are larger; both when it comes to using the same test environments and the same or similar test methods for the two concerns.

A reason why synergies are abundant in verification is that the bulk of testing required by the standards is not safety- or cybersecurity specific per se but rather aimed at ensuring good product quality in general, for instance requirements, resource and robustness testing. Except for a few more test methods necessary to cover both standards, the main difference is the need to test specifically added safety- or security mechanisms that are not part of performing the nominal function, and even when testing these mechanisms, many of the test methods are the same or similar.

2 Background

The analysis of potential synergies is based on a case study conducted as part of AMASS [5], a research project which aims to develop tools and methods targeting different aspects of assurance and certification of cyber-physical systems, including multi-concern assurance. An assurance case needs to communicate the scope of the system, the operational context, the claims, the arguments to give the rationale for the claims, along with the corresponding evidence. A multi-concern assurance case, however, should support this for more than one quality attribute. The multi-concern assurance approach employed here has a tool-supported methodology for the development of assurance cases which address multiple system characteristics and provide exploitable synergies between them [6].

In this section, we present the use case studied via a palette of open-source tools. In particular Papyrus [7] and SysML have been used to model the system, context and requirements. OpenCert [8] is used to manage process modelling and to argue for a multi-concern assurance case. It is the process modelling and the subsequent analysis of complement of development activities that has provided most of the insights to co-engineering findings presented in this paper.

2.1 Use Case

The use case studied in this paper is a positioning component for automated driving systems (ADS) and needs to conform to both functional safety and cybersecurity. The positioning component can be used in various functions and is designed as an element-out-of-context (EooC). Here we generalize the term safety-element-out-of-context (SEooC) used in ISO 26262 to apply for any quality attribute. In an EooC, the requirements (in this case functional, safety and security requirements for the positioning unit) are based on an assumed context at the design time on its use. The assumptions must later match the requirements in any real context where the component is actually used. The component is aimed at automotive functions; therefore ISO 26262 is used as safety standard and a first working draft of ISO/SAE 21434 for cybersecurity.

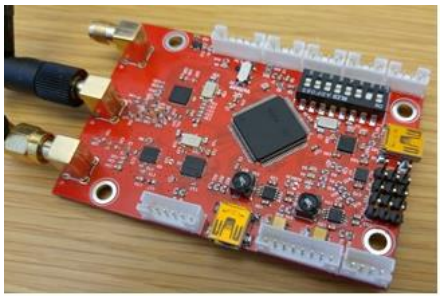

Fig. 1. Hardware for positioning element-out-of-context.

Fig. 2. Scale model of an Automated drive capable vehicle used for testing.

Fig. 1 shows the hardware for the positioning element. It contains a satellite navigation receiver which is used in conjunction with correction data for enhanced precision (real-time kinematic positioning). The correction data is streamed over an Internet connection. Together with data from an inertial measurement unit (IMU) and odometry a position can be calculated, with a quality-measure of that position [9].

To complete the use case, the positioning component is matched to the hypothetical context of an ADS function named Automated driving controller (ADC), which is a highway autopilot. Fig. 2 shows the vehicle level test environment for the ADC function. A detailed description of the function is beyond the scope of this paper; however, the function has a number of functional requirements which are analyzed for safety and security risks according to both standards, resulting in additional safety goals (top-level safety requirements) and security goals. A simplified definition of one of the functional requirements is: *The automated driving mode of the ADC function may only be activated on roads certified for ADS vehicles, and only when an enable signal is received from a road-side unit.*

The hazard analysis and risk assessment (HARA) required in ISO 26262 results in safety goals for the ADC. A safety goal related to the stated functional requirement is: *ADC may only be activated on certified roads, corresponding to an Automotive Safety Integrity Level D (ASIL D)*. ASIL D is the highest integrity level requiring the most stringent measures to avoid a failure. Since the function is only designed to work within the parameters given in the functional requirement, its behavior is undefined if enabled anywhere else, thus resulting in high risk of harm. Consequently, a high level of risk reduction is necessary.

For cybersecurity, a threat analysis and risk assessment (TARA) is used to elicit security goals. A security goal with a dependency to the mentioned safety goal is: *Methodically designed and tested integrity protection to fulfill ADC may only be activated on certified roads*, corresponding to Cybersecurity Assurance Level 3 (CAL 3). CAL 3 is the second highest risk reduction level according to the first working draft of ISO/SAE 21434. When the safety goal is refined in the design phases, it results in safety requirements on the positioning of the vehicle, i.e. that the position can be matched to a map making sure the vehicle is driving on a certified road. Moreover, there will be corresponding security requirements on the integrity of both positioning

and the certified-road open for traffic signal data. This is since the data from the wireless communication interfaces are susceptible to be compromised by an adversary.

3 Multi-Concern Development Lifecycle

In this section, we present our findings based on a comparison between two approaches for analyzing a multi-concern development lifecycle, namely *separate* and *co-engineered*. Here the separate approach means preforming the additional work of satisfying a quality attribute together with the nominal implementation but separate from any additional quality attribute. This would be comparable to carrying out the fulfillment of the quality attributes in sequence, or in parallel with different development teams. In this paper, this means that functional safety is handled first by complying to the ISO 26262 standard, and then cybersecurity is tackled by complying to a first working draft of ISO/SAE 21434 standard. In the co-engineered approach, on the other hand, quality attributes are analyzed in parallel, which according to our use case, could be interpreted as the co-engineering of safety and security.

Our findings on the multi-concern development synergies are divided into *co-design* (see Sec. 3.1) and *co-verification* (see Sec. 3.2), referring to the left side and the right side of a V-model development lifecycle, respectively. Fig. 3 illustrates the lifecycles for nominal function with added activities for functional safety and cybersecurity.

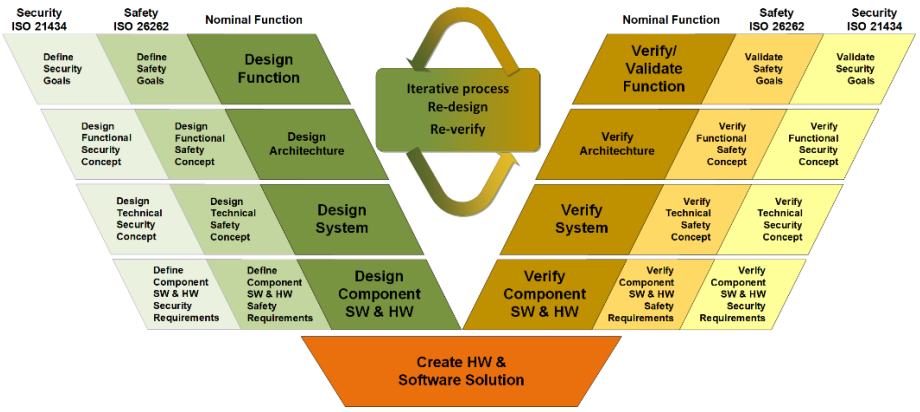

Fig. 3. Co-engineering of automotive Safety (ISO 26262) and Security (ISO/SAE 21434)

3.1 Co-Design

In the design phases, the overlap, and hence potential for synergies between safety and cybersecurity in terms of analysis methods, countermeasures and requirements were found to be limited in our use case. For this part of the development lifecycle, the main advantage of treating both concerns in parallel is completeness, i.e. that both concerns and their interplay are considered through all stages. Even though this does

not reduce the work of adhering to two standards, it contributes to reducing the risk of missing issues that may cause costly major redesigns if discovered later, or even worse, resulting in too high residual risks if dependencies are not discovered at all.

Risk Analysis. In this paper, the risk analysis is handled separately for the two concerns through hazard analysis and risk assessment (HARA) and threat analysis and risk assessment (TARA). However, we perform the activities needed to analyze risks simultaneously and coordinated, facilitating co-analysis of risks for the different concerns. The only activity missing for a complete co-analysis is trade-off analysis, corresponding to the way that different concerns impact each other. Specific details and information how to perform the risk analysis is outside the scope of the paper.
A significant difference between HARA and TARA is the comprehensiveness of the organization level inputs for the latter. These inputs include e.g. assets and their values, stakeholders, and threat information. When doing the asset identification on the organizational level the dependency between security and safety becomes obvious, if safety is a concern then it automatically becomes an asset for the organization. This dependency will carry over to the implementation of the item if it is deemed to be security related. It is the absence of an architecture to be analyzed in the early stages in the development process that drives the need for a rigorous asset identification to facilitate the dependency identification.

Table 1 shows the HARA performed according to the ISO 26262 standard and Table 2 shows the TARA performed according to the ISO/SAE 21434 standard. Moreover, the HARA and TARA analyses result in the goals specified in Table 3. The analyses are performed on the function level, i.e., independent of the architecture. This enables the dependency identification to be performed on early stages of the development lifecycles and distinguishes our approach from earlier approaches such as SAHARA [10] or the one proposed by Schmittner et al. [11].

Requirements. Co-requirement-engineering have exposed dependency relationships between attributes. Often the safety-requirements depend on the fulfilment of security-requirements. Table 3 shows one such example, where the cybersecurity goal references the safety goal. This means that the safety goal can be violated if the cybersecurity goal is violated, or in other words that the desired safety risk reduction cannot be achieved unless the security risk reduction is also achieved. The refined requirements allocated to the architecture will retain the dependency inherited from the original goals. While it is beneficial to have these dependencies explicit at an early stage in the development process from a completeness viewpoint, no labor-saving synergy effect was encountered in the analysis or requirements engineering process.

Countermeasures. Except for the often-mentioned encryption measure that can be used to safeguard confidentiality in terms of security, and at the same time protect against data corruption and in terms of safety, no other measures have been found so far to contribute to the co-engineering of implementation of mechanism.

Table 1. Excerpt of HARA for our defined Safety Goal.

Function	ID	Hz_ADC_019
	Use case	Normal driving
	Driver	Human
Failure mode (guide words)	Failure mode	Commission
	Effect	Fully
Situation Description	Driver	Human
	Location	Country road
	Vehicle speed	Medium, High
	Manoeuvre	Steering
	Traffic intensity	Any
	Environmental changes	No change
	Persons at risk	Any
	Target	Off road, VRU, motorcycles, cars, trucks, stationary object
Hazard Description	Failure	Unintended activation of ADC
	Failure effect	Decreased curvature
	Amplitude	High
	Duration (ms)	High
	Hazard event	Head-to-head collision
Exposure	E	**E4**
Severity	S	**S3**, Nominal ADC function is not intended outside of certified road. ADC behaviour is not predictable, thus worst-case situation is assumed
Controllability	C	**C3**, ADC is in control, thus no human controllability

Table 2. Excerpt of TARA for our defined Security Goal.

Asset tag for item	ADC_ATAG8
Owner	Safety Manager
Primary assets	IS026262 (data)
Security property	Integrity
Attack path (subset wireless)	Wireless
System specific	Yes
Safety	4
Privacy	0
Financial	0
Operational	2
S (max)	4
Elapsed Time (dependent on controls)	7
Expertise	6
Knowledge	3
Window of Opportunity	4
Equipment	4
Attack potential required (in numbers)	24
Attack potential required to exploit scenario	High
Likelihood	Unlikely
Cybersecurity Assurance Levels	CAL3

Table 3. Safety and cybersecurity.

Safety Goal	ASIL	D
	SG ID	SG_ADC_001
	SG	ADC may only be activated on certified roads.
	Safe state	ADC disabled (Human in control of the vehicle)
	FTTI	150 ms
Cybersecurity goal	***goal of attack resistance*** *(of each threat/asset pair) to be considered during design and testing* Methodically designed and tested integrity protection to fulfil *"ADC may only be activated on certified roads"*	

3.2 Co-Verification

For the right side of the V-model, i.e. the verification and validation side, the possible synergies are larger, for the use case studied in this paper. This includes synergies related to the test environments and the test methods for the two concerns. A reason why synergies are abundant in verification is that the bulk of testing required by the standards is not safety- or cybersecurity specific per se but rather aimed at ensuring good product quality in general. Except for a few more test methods necessary to cover both standards, the main difference is the need to test specifically added safety- or security mechanisms that are not part of performing the nominal function, and even when testing these mechanisms, many of the test methods are the same or similar.

There are three major areas corresponding to the right side of the combined development lifecycle that have been identified to benefit from co-engineering. These areas include *test environments* and the *test purposes* for each environment and the *test techniques* employed to fulfill these purposes (see Fig. 4). Different maturity of the implementation is tested using model-in-the-loop (MIL), software-in-the-loop (SIL), and hardware in-the-loop (HIL).

Test Environments. As illustrated in Fig. 4, test environments could be mapped to different integration levels, namely component level (test environment 1), system/subsystem level (test environment 2) and complete vehicle level (test environment 3 and 4, see Fig. 2). These environments are fully re-usable in terms of testing for the different concerns, i.e., nominal function, security and safety. This is a major benefit, compared to building, maintaining, and operating separate testing environments. The test environments can be used for regression testing, which is a prerequisite for continuous deployment needed to meet the maintenance requirements of security. The test environments can also be used for back-to-back testing if model-driven development is used.

Test Purposes or Test Goals. Fig. 4 also shows the purpose of tests, which could be classified into one of the following categories:

- Correctness of Implementation of Specification
- Robustness

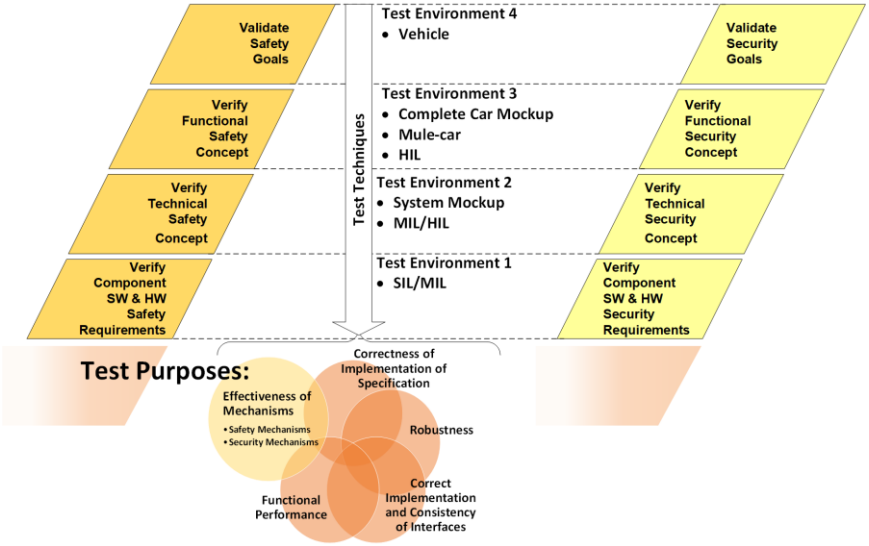

Fig. 4. Test techniques and test purposes at different integration levels and environments.

- Correct implementation and consistency of interfaces
- Functional performance, accuracy and timing
- Effectiveness of mechanisms

The above categories aim to detect systematic faults during the different levels of integration. According to the use case studied in this paper, all mentioned test purposes (except the *effectiveness of mechanisms*) show a large overlap between the different concerns. This facilitates co-verification for both test environments and test purposes. Note that, when it comes to the "effectiveness of mechanisms", the overlap is dependent on parameters such as the competence of testers and the type of mechanisms used. In fact, previous studies [12] have shown that different safety mechanisms could impact system security both negatively/positively[1]. The same conclusions are drawn in the same study about the impact of security mechanisms on system safety[2]. This means that even in the case of the "effectiveness of mechanisms" test purpose, the overlap could be increased by improving the testers' competence as well as by choosing mechanisms that simultaneously provide both safety and security when possible.

Test Techniques. Test techniques are also spread throughout different integration levels. These techniques could be classified into *analytical* and *experimental* techniques. The former type of techniques corresponds to a cost-effective way to perform verification and validation of safety and security based on mathematical techniques or

[1] Table 4-1 in the Interplay Between Safety and Security deliverable of HEAVENS project [12]
[2] Table 4-2 in the Interplay Between Safety and Security deliverable of HEAVENS project [12]

models. Examples of these models include, fault/attack trees, markov models, and stochastic petri nets. Experimental test techniques, however, make use of different testing systems to reveal deficiencies and obtain measures to grade the systems' safety or security capabilities. Examples of test techniques and their corresponding test purposes are presented in Table 4.

The overlap between different test techniques aiming to achieve a specific test goal varies significantly for different concerns. Similar to the discussion on test purposes, testers' competence is one of the main factors affecting the overlap between the work that needs to be done to address different concerns. For example, error guessing and penetration testing techniques (see Table 4. are both dependent on field data as well as security expertise, respectively. This indicates that by educating testers that could work on multiple concerns, the overlap could be increased. Moreover, even when the overlap is small, the lessons learned from pursuing a test purpose when investigating system safety could be reused when addressing security or vice versa. For example, fault injection is a well stablished testing technique used to evaluate system safety. Lessons learned from the past works that used this technique could be used to inject attacks instead of faults to evaluate system security [13]. This is due to the fact that security attacks may be considered as a special type of faults which are human made, deliberate and malicious, affecting hardware/software from external system boundaries and occurring during the operational phase [14].

4 Conclusions and Future Work

In this paper, our goal was to study and compare two approaches on multi-concern development, namely *separate* and *co-engineered*. In the former, quality attributes (concerns) are analyzed in sequence, whereas in the latter, the attributes are analyzed in parallel.

Table 4. Examples of test techniques that could be used to achieve different test purposes.

Test purpose	**Test technique**
Correctness of Implementation of Specification	Requirement based test
	Back-to-back test
	Fault injection test
Robustness	Stress test and resource usage test, environmental test
	Long term test and user test
Correct implementation and consistency of interfaces	Test of internal and external interfaces
	Full Communication test compatibility and timings
	Fuzz testing
Functional performance, accuracy and timing	Performance test
	Back-to-back test
Effectiveness of mechanisms	Fault injection test
	Attack injection test
	Error guessing
	Penetration testing

The quality attributes analyzed are functional safety and cybersecurity; and the analysis is performed by following the ISO 26262 standard (addressing functional safety) and the first working draft of the ISO/SAE 21434 standard (addressing cybersecurity).

Our analysis shows that for the design phases, i.e. the left side of the development V-model used in both standards, the overlap between safety and cybersecurity is small. However, the parallel analysis of safety and cybersecurity results in an improvement of completeness, thereby reducing the risk of missing issues that may cause costly major redesigns if discovered later. For the right side of the V model, i.e. the verification and validation side, the possible synergies are larger; both when it comes to using the same test environments and the same or similar test methods for the two concerns.

The analyses performed in this paper – and hence the results obtained – are based on a use case developing a positioning component for the automotive domain. And the search for synergies was carried out, without any tailoring of the development life-cycle, or any real special methods and tool geared towards co-engineering. This was done to act as a baseline for future work. In the creation and argumentation for a complete multi-concern assurance case, it is expected that the tools mentioned in Sec. 2 will really come into their own. There is also an opportunity employ new multi-concern methods, that combined with tool support is expected deliver more substantial synergies.

Acknowledgements. This work is supported by the EU and VINNOVA via the ECSEL Joint Undertaking project AMASS (No 692474), but the contents of the paper only reflect the authors views.

References

1. ISO 26262:2011 - Road vehicles -- Functional safety, https://www.iso.org/standard/43464.html.
2. IEC 61508:2010 - Functional safety of electrical/electronic/programmable electronic safety-related systems, http://www.iec.ch/functionalsafety/.
3. Gallina, B., Sljivo, I., Jaradat, O.: Towards a Safety-Oriented Process Line for Enabling Reuse in Safety Critical Systems Development and Certification. In: 2012 35th Annual IEEE Software Engineering Workshop. pp. 148–157 (2012).
4. ISO/SAE AWI 21434 - Road Vehicles -- Cybersecurity engineering, https://www.iso.org/standard/70918.html.
5. About | AMASS, https://www.amass-ecsel.eu/.
6. Ruiz, A., Gallina, B., de la Vara, J.L., Mazzini, S., Espinoza, H.: Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems. In: Skavhaug, A., Guiochet, J., Schoitsch, E., and Bitsch, F. (eds.) Computer Safety, Reliability, and Security. pp. 311–321. Springer International Publishing, Cham (2016).
7. Papyrus, https://www.eclipse.org/papyrus/.
8. Beaton, W.: OpenCert, https://www.polarsys.org/projects/polarsys.opencert.

9. Vedder, B., Vinter, J., Jonsson, M.: Accurate positioning of bicycles for improved safety. In: 2018 IEEE International Conference on Consumer Electronics (ICCE). pp. 1–6 (2018).
10. Macher, G., Sporer, H., Berlach, R., Armengaud, E., Kreiner, C.: SAHARA: A security-aware hazard and risk analysis method. In: 2015 Design, Automation Test in Europe Conference Exhibition (DATE). pp. 621–624 (2015).
11. Schmittner, C., Gruber, T., Puschner, P., Schoitsch, E.: Security Application of Failure Mode and Effect Analysis (FMEA). In: Bondavalli, A. and Di Giandomenico, F. (eds.) Computer Safety, Reliability, and Security. pp. 310–325. Springer International Publishing, Cham (2014).
12. HEAVENS, https://www.sp.se/en/index/research/dependable_systems/heavens/Sidor/default.aspx.
13. B. Sangchoolie, P. Folkesson, J. Vinter: A Study of the Interplay Between Safety and Security Using Model-Implemented Fault Injection. in EDCC 2018: 14th European Dependable Computing Conference, 2018.
14. Avizienis, A., Laprie, J.-C., Randell, B., Landwehr, C.: Basic concepts and taxonomy of dependable and secure computing. IEEE transactions on dependable and secure computing. 1, 11–33 (2004).