 doi:10.20944/preprints202011.0644.v1

Article

Towards Security-by-design in Automotive Development Process

Seungyeon Jeong[1], Sooyoung Kang[2], Seungjoo Kim*

[1] Department of Automotive Convergence, Korea University, Seoul 02841, South Korea; sodon513@gmail.com
[2] Center for Information Security Technologies, School of Cybersecurity, Korea University, Seoul 02841, South Korea; bbang814@gmail.com
* Correspondence: Center for Information Security Technologies, School of Cybersecurity, Korea University, Seoul 02841, South Korea; skim71@korea.ac.kr

Abstract: Although traditional automotive development has mainly focused on functional safety, as the number of automotive hacking cases has increased due to the growing Internet connectivity of vehicles, security is also becoming more important. Accordingly, international organizations are preparing regulations to ensure security on automotive development by emphasizing the concept of security-by-design. The problem, however, is that no specific methodology has been suggested. In this paper, we propose specific security-by-design methodology for automotive development based on standards related to security-by-design. With our methodology, automotive manufacturers can consider aspects of trustworthiness, and can also respond to the upcoming cybersecurity regulation.

Keywords: Automotive development, Secure SDLC, Evidence-based standard, ISO/SAE 21434, UNECE cybersecurity regulation

1. Introduction

The introduction should briefly place the study in a broad context and highlight why it is important. It should define the purpose of the work and its significance. The current state of the research field should be reviewed carefully and key publications cited. Please highlight controversial and diverging hypotheses when necessary. Finally, briefly mention the main aim of the work and highlight the principal conclusions. As far as possible, please keep the introduction comprehensible to scientists outside your particular field of research. References should be numbered in order of appearance and indicated by a numeral or numerals in square brackets, e.g., [1] or [2,3], or [4–6]. See the end of the document for further details on references.

Unlike functional safety, security was not a focus in automotive development. Security is a concept that includes confidentiality which ensures authorized users only have access to information assets of the system, integrity which ensures the system is fully preserved without inappropriate change or destruction of information and availability which ensures access to and use of system information at any time users want [4]. Security aims to prevent a situation where external security threats expand into the system and cause damage to users and if it is not guaranteed, it could lead to various accidents such as loss of life or privacy. Recently, with the advent of connected cars, the software proportion and internet connectivity of vehicles are growing, and the possibility of vehicles being exposed to security threats is increasing accordingly [5], [6]. As a result, the necessity of automotive security development is rising, and international organizations are showing efforts to emphasize it by enacting automotive cybersecurity regulations. Especially, the UNECE automotive cybersecurity regulation (UNECE regulation) will be applied from 2022 based on new vehicles, and according to this, vehicles that have not been evaluated and certified with the regulation will not be allowed to be exported to Europe [7]. Therefore, developing secure vehicles is an important issue not

only for various security threats but also for the automotive import and export economy that will be faced right away.

The UNECE regulation proposes security-by-design as a core requirement, which is the concept of implementing a trustworthy product by considering all factors of functional correctness, safety, and security from the beginning of product development. In particular, since automotive development has a long life cycle and complex supply chain, it very difficult to change the architecture after development. Therefore, security-by-design must be dealt with more importantly in automotive development, and it can be achieved by the Secure System Development Life Cycle (secure SDLC). Secure SDLC is a systematic security development framework that is applied throughout the entire product development life cycle. It is used by many companies (e.g. Microsoft) or standard organizations (e.g. National Institute of Standards and Technology (NIST)), and relevant studies are also actively carried out [8] – [15].

However, the existing secure SDLC standards do not pro-vide an overall and specific methodology for the security-by-design in automotive development, since they not only target software mostly but also emphasize different aspects of activities, such as eliciting systematic requirements or acquiring third-party components. In addition, the aforementioned UNECE regulation does not provide a specific methodology of achieving security-by-design, and it is the same for relevant studies.

Therefore, in this paper, we propose Trustworthy Automotive SDLC as a specific methodology for security-by-design in the automotive development process. Trustworthiness is a concept providing the trust that the system will operate as we expected by considering all the functional correctness, safety, and security of the system in development [16]. Trustworthiness should be particularly emphasized in the system where functional safety is important such as the automotive system.

In order to propose Trustworthy Automotive SDLC, we firstly derive activities related to automotive development from 4 major secure SDLC standards. These include Microsoft Security Development Lifecycle (Microsoft SDL), NIST Secure System Development Life Cycle (NIST SSDLC), The Open Web Application Security Project Comprehensive, Lightweight Application Security Process (OWASP CLASP) and Society of Automotive Engineers J3061 (SAE J3061). Afterward, each activity is mapped to the detailed contents of evidence-based standards to derive Trustworthy Automotive SDLC in detail. Evidence-based standards are standards composed of detailed requirements on performing an activity of a process. Since verifying the source of collected evidences, these ensure traceability between each evidence and each phase. In this paper, we consider 4 number of evidence-based standards: CC (Common Criteria, ISO/IEC 15408) which is a standard related to security evaluation of IT products, ISMS (Information Security Management System, ISO/IEC 27001) which is a standard related to security evaluation of development environment, PIMS (Privacy Impact Management System, ISO/IEC 27701) which is a standard related to privacy and FSMS (Functional Safety Management System, ISO 26262) which is a standard related to automotive functional safety. Especially CC is very useful when you want to build secure SDLC since it specifies requirements for documents to be produced. Also, the requirements of the UNECE regulation and the ISO/SAE 21434 international automotive cybersecurity standard which is the basis of the UNECE regulation are applied to Trustworthy Automotive SDLC in this procedure.

Trustworthy Automotive SDLC is easy to be integrated into the existing functional safety development process since it is based on the process of ISO 26262 and takes into account all aspects required in automotive development: functional correctness, safety, and security. In addition, Trustworthy Automotive SDLC ensures competitiveness against security threats by providing a sufficient security level required in automotive development and suggests detailed activities to utilize it in upcoming UNECE regulation.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 November 2020 doi:10.20944/preprints202011.0644.v1

2. Related Work

2.1. Relevant studies

To emphasize the need for Trustworthy Automotive SDLC, we analyzed relevant studies on automotive security development. We have chosen to include all publications on secure SDLC in the last 10 years (2010 to 2020) from 4 number of well-known digital libraries: ACM, IEEE, Springer, Elsevier. In addition, to classify a publication as relating to secure SDLC, we decided some essential keywords. 1) 'Automotive' or 'CPS (Cyber Physical System)', 2) process-related keywords (e.g. 'process', 'life cycle', 'development'), phase-related keywords (e.g. as 'requirements') or activity-related keywords (e.g. 'fuzz testing').

We collected 65 studies [17] – [41] and Figure 1 is a graph classifying studies based on the phases covered by each study. As a result of analyzing relevant studies, it has shown that most of the studies were focused primarily on requirements and design phase. Some have proposed model for the entire development process, but they were only conceptual approaches and did not provide detailed activities [18], [19], [22], [29], [30]. Therefore, with the analysis result, it is possible to feel the need for a detailed methodology for automotive security-by-design throughout the entire development process.

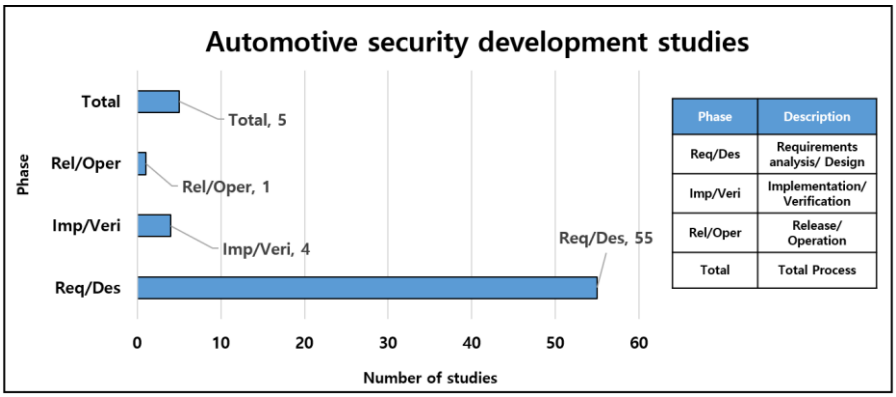

Figure 1. Automotive security development studies.

2.2. Secure SDLC standards

In the case of existing product development, security has been improved through secure SDLC [42]. Secure SDLC enables the development of secure products by considering security-related activities for all phases. However, the existing secure SDLC standards do not provide overall and sufficient details of the actual application, especially in the case of automotive development. Therefore, we establish universal security-by-design methodology by integrating existing secure SDLC standards. There are 4 number of target secure SDLC standards: Microsoft SDL based on the software [43], NIST SSDLC based on the system [44], OWASP CLASP based on enterprise best practices [45], SAE J3061 based on the vehicle [46]. Since each secure SDLC standard emphasizes different aspects such as deriving systematic security requirements or acquiring third-party components, overall activities for the development process can be derived by integrating them. Table 1 shows the features of each standard.

2.3. Automotive cybersecurity regulation and standard

With the advent of the connected car, the portion and connectivity of automotive software increases, and the importance of automotive security is growing. Accordingly, various international organizations are enacting regulations to ensure the security of automotive development as we mentioned earlier [47]. Especially, UNECE is enacting automotive cybersecurity regulation to ensure security throughout the entire development process, which will be effectuated from 2022 on new vehicles and 2024 on existing vehicles [48]. UNECE regulation is based on the ISO/SAE 21434. ISO/SAE

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 November 2020 doi:10.20944/preprints202011.0644.v1

21434 is an international standard for automotive cybersecurity established by ISO based on SAE J3061 and it will also be published in 2022 [49], [50]. Therefore, we propose Trustworthy Automotive SDLC that covers all the requirements of both UNECE regulation and ISO/SAE 21434.

Table 1. Features of secure SDLC standards.

	Microsoft SDL	NIST SSDLC	OWASP CLASP	SAE J3061
Target	Software	System	Best practices	Vehicle
Feature	Provide the developer-oriented process	Focuses on third-party components acquisition and system disposal	Provides real-world enterprise activities in the form of best practices	Provides rough activities of secure automotive development
Pros&Cons (P: Pros, C: Cons)	Provides tools for performing activities such as risk analysis (P) Not include disposal phase (C)	Used to evaluate systems that require certification & accreditation (P) Lack of documented information (C)	Identifies the role of the personnel in charge of each activity (P) Lack of information because the project period expired (C)	Easy to apply security-related activities according to the automotive function safety development process (P) Not include training and disposal phase (C)

3. Trustworthy Automotive SDLC

In this chapter, we establish universal security-by-design methodology encompassing the existing secure SDLC standards and, derive Trustworthy Automotive SDLC suitable for automotive development based on it. Figure 2 shows the procedure of deriving the Trustworthy Automotive SDLC. At first, we extracted the activities related to automotive development from 4 number of secure SDLCs which present only somewhat rough activities. Then with 4 number of evidence-based standards, we extracted detailed items to comply with standards of each field (such as detailed activities to be performed or outputs to be derived). Also, we pulled out the requirements which are essential for automotive development. As a result, the activities related to the automotive development process were further detailed through detailed items of the standard of each field and filtered based on the requirements derived in this study, and finally, Trustworthy Automotive SDLC was derived.

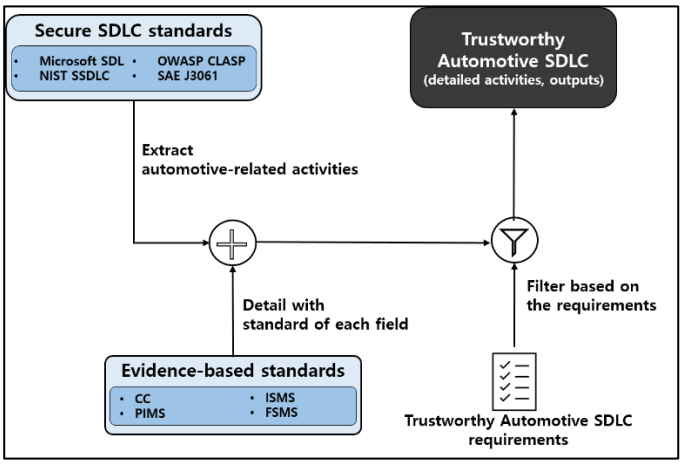

Figure 2. Procedure of Trustworthy Automotive SDLC construction.

This is the development process suggested from the view of developing the whole vehicle system and by considering both functional safety and security, it enables the development of a trustworthy vehicle. Trustworthy Automotive SDLC also enables developing a product that is more secure than traditional development methodologies which considered security with only a few activities, such as penetration testing. The subsections describe the requirements, design methodology, activities of each phase and detailed activities and outputs of each phase.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 November 2020 doi:10.20944/preprints202011.0644.v1

3.1. Requirements

The requirements of Trustworthy Automotive SDLC are as follows:

1. Targets on system
2. Includes the procedure of third-party components acquisition
3. Considers the aspect of trustworthiness
4. Satisfies the level of safety and security required for automotive development
5. Ensures traceability between phases
6. Provides detailed activities for every phase of the process (depth = 2)

First, vehicles are developed in the form of a system that includes software and hardware [51]. Therefore, Trustworthy Automotive SDLC must build a system-based process. Also, automobile manufacturers do not develop all the components of vehicles by themselves but utilize components acquired from tier-1 or tier-2 suppliers [52]. Therefore, in order to attain trustworthiness for third-party components, Trustworthy Automotive SDLC must perform activities for obtaining third-party components.

Thirdly, trustworthiness which includes all aspects of functional correctness, safety, and security should be considered in automotive development [53]. Since functional correctness and safety have been considered through the functional safety development life cycle and safety level Automotive Safety Integrity Level (ASIL) suggested by ISO 26262, we need to design a methodology that can combine security based on them. In particular, the portion where functional correctness, safety, and security goals that conflict should be identified in the early phase of the development process [54].

The ASIL required by ISO 26262 is selected differently depending on the function of the vehicle, and the automotive functional safety development process develops a vehicle based on the ASIL assigned to each function. According to U.S. security company Synopsys, the functional safety development process should satisfy ASIL C on average for core functions [55]. With respect to security, 2 sides need to be considered. Firstly, the EAL required for the automotive system should be met. The EAL is the assurance level of CC, an international standard related to IT product security evaluation. According to [56], the ASIL C of ISO 26262 corresponds to the EAL5 of CC. Therefore, in this paper, we determine that Trustworthy Automotive SDLC should cover the EAL5 to ensure sufficient security of automotive development. In addition, Trustworthy Automotive SDLC should also reflect the requirements of automotive cybersecurity regulation which is essential for automobile manufacturers targeting not only domestic but also overseas markets. Thus, we consider UNECE regulation and ISO/SAE 21434, as mentioned earlier, and for ISO/SAE 21434 only the essential requirement RQ (Requirement) is considered.

Since the vehicle is a critical system in which a system problem can lead to human casualties [52], [57], Trustworthy Automotive SDLC should ensure traceability of the system more rigorously by verifying the consistency and completeness between goals, requirements, architectures, and implementation according to phases. In addition, Trustworthy Automotive SDLC should make it easy to apply itself to the real-world by proposing a sufficiently detailed methodology for every phase. In this paper, the meaning of sufficient detail is defined by depth. Depth indicates the activity level of detail defined in this paper, and it can be graded for each activity. For example, phase such as verification has a depth of 0, sub-phase such as testing has a depth of 1, and activity such as static analysis has a depth of 2. Trustworthy Automotive SDLC we propose has a format that can be applied to the actual product since it provides the activities of all phases to the depth of 2 so that we can directly apply them to the detailed items of evidence-based standards.

3.2. Framework design methodology

This section describes the methodology of deriving Trustworthy Automotive SDLC based on the requirements of Section 3.1. In this paper, we selected 4 number of representative secure SDLC standards in each domain (functional safety, security, privacy) and derived activities of each phase in order to build universal security-by-design methodology. Subsequently, all activities were

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 November 2020 doi:10.20944/preprints202011.0644.v1

mapped to each content of evidence-based standards. There has been a steady stream of studies mapping the development process to multiple evidence-standards, but they were all focused on partial phases (e.g. [58] for requirements analysis phase) or not sufficiently specified ([59] – [68]). In this paper, contents of 4 number of evidence-based standards (CC, ISMS, PIMS, and FSMS) were mapped to each activity of the Trustworthy Automotive SDLC phase and refined each activity based on this. Table 2 shows the features of evidence-based standards we selected. With this procedure, Our Trustworthy Automotive SDLC can further refine all activities in all aspects of security, privacy, and functional safety for products and product development environments. Figure 3 shows mapping relation between our Trustworthy Automotive SDLC and evidence-based standards. Lastly, activities that satisfy the requirements of Section 3.1 are selected to derive Trustworthy Automotive SDLC. The finally derived Trustworthy Automotive SDLC is as shown in Figure 4 and it consists of a total of 50 detailed activities for 10 phases.

The proposed Trustworthy Automotive SDLC, as mentioned earlier, is a system-targeted methodology that performs the acquisition procedure for third-party components through the acquisition phase. Also, as shown in Figure 3, it is possible to map all phases with each part of FSMS, so it can be merged into the existing functional safety development process which follows ISO 26262. Also, as I mentioned before, goals of functional safety and security could conflict while merging the security development process and the functional safety development process, and it can be resolved by activities such as 3.1.2 conformity & conflict check on impact assessment results or 3.2.2 conformity & conflict check on requirements. Moreover, our Trustworthy Automotive SDLC consists of detailed activities satisfying the security level required for automotive development (which is EAL5) and satisfying requirements of UNECE regulation and ISO/SAE 21434.

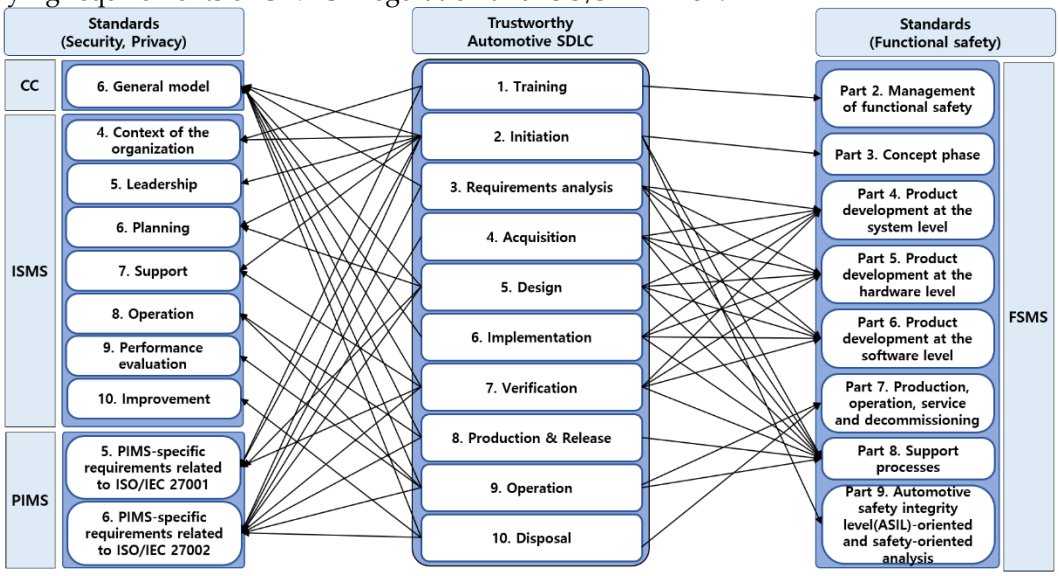

Figure 3. Mapping between Trustworthy Automotive SDLC and evidence-based standards.

3.3. Activities for each phase

3.3.1. Training phase

During the training phase, organization members gain the awareness of safety, security, and privacy, and get training about the relevant knowledge required as they go through the development process. Teams developing or managing functions related to security or safety (e.g. security team) have background knowledge, but most other teams (e.g. development team) often lack knowledge of trustworthiness. Therefore, it is necessary to train the subjects required for process execution through training by domain. Basic training covers topics for gaining awareness of safety, security, and privacy. In addition, by training basic knowledge on topics covered in the development process (e.g. risk analysis), it ensures that organization members do not feel uncomfortable in carrying out the process. Advanced training is an additional training which trains how to carry out the detailed activities of Trustworthy Automotive SDLC. It is given to personnel according to their task. From a

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 November 2020 doi:10.20944/preprints202011.0644.v1

security perspective, this includes topics such as security design and secure coding for the development team, static and dynamic analysis for evaluation team, security, and privacy-related regulatory certification for security team. All training is managed by the road map, and by managing compliance with training according to the target audience, traceability of training is ensured.

3.3.2. Initiation phase

In the initiation phase, the development environment for the project is established, overall plans of the project are established, and goals for each domain are set. First, this phase carries out project categorization by considering the information assets, product types, and project characteristics handled by the project. In addition, based on them, plans for the entire project including roles, tools, and minimum quality level are established. Particularly, minimum quality level is the minimum level of security, safety, and privacy that must be met by the project. If it is not satisfied, the following phases cannot be carried out. The initiation phase also establishes goals by domains and verifies consistency and completeness between goals to ensure traceability to the performance of subsequent phases.

In the case of the automotive system, the development is performed separately according to the system, software, and hardware-level. Therefore, the corresponding phase should be performed according to each level. At the initiation phase on system-level, consistency of software and hardware development should be ensured by establishing the overall environment and plans covering the rest of the levels (i.e. hardware and software). As shown in Figure 4, this configuration is equally applied from the initiation to the production & release phase.

3.3.3. Requirements analysis phase

In the requirements analysis phase, impact assessment is performed on the project's security, safety, and privacy-related assets, and based on the results, requirements by domains are derived. This phase also ensures traceability between the initiation phase and the requirements analysis phase by verifying consistency and completeness between goals and requirements. As mentioned earlier, goals of safety and security can conflict [54]. Therefore, this phase identifies priority for safety and security impact level through the activity of 3.1.2 conformity & conflict check on impact assessment results. At this point, security and safety impact on the reused existing system is also evaluated. Also, conformity and conflict of requirements by each domain are also checked by the activity of 3.2.2 conformity & conflict check on requirements. Lastly, by verifying the consistency and completeness of requirements according to goals previously derived, traceability between the initiation phase and requirements analysis phase is ensured.

3.3.4. Acquisition phase

At the acquisition phase, the scope and plans of third-party component acquisition are established and the relevant requirements are defined. Also, based on the requirements, evaluation and test are performed on the specifications of third-party components. Third-party components are those acquired by tier-1 or tier-2 suppliers. In addition, in this phase, automobile manufacturers perform independent evaluations and tests based on the specifications of third-party components submitted in accordance with the automobile manufacturer's requirements. In particular, in the case of automotive development, the development of subsystems through partners plays a significant role in overall development, so it is essential to perform the corresponding phase.

3.3.5. Design phase

During the design phase, the architecture is designed and risk analysis for each domain is performed based on it. Since automotive development constructs one integrated system based on lots of subsystems, the corresponding phase considers the system integration procedure. Also, based on the integrated architecture, risk analysis is performed by domain to derive mitigation for possible threats of the system. As with the preceding phase, the design phase checks the conformity and

 doi:10.20944/preprints202011.0644.v1

conflict of domain-specific mitigation so that there is no conflict between the functional safety development process and the security development process. It also ensures traceability between the requirements analysis phase and the design phase by verifying the consistency and completeness of

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 November 2020 doi:10.20944/preprints202011.0644.v1

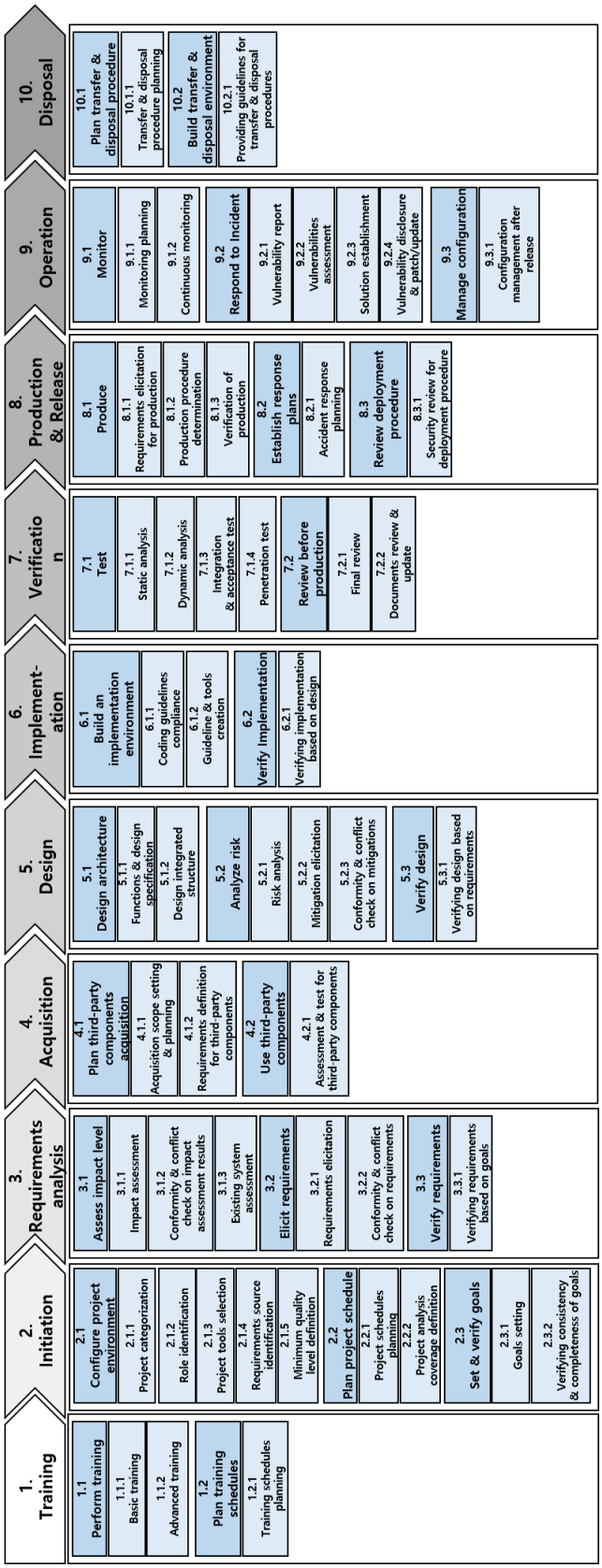

Figure 4. Trustworthy Automotive SDLC.

 doi:10.20944/preprints202011.0644.v1

the architecture following the requirements.

3.3.6. Implementation phase

In the implementation phase, the system is implemented based on the requirements and architectures derived ahead. At this phase, development team implements the system with coding guidelines, which may include previously established coding standards such as MISRA C [69]. The development team also creates deployment guidelines or tools to enable users to build a trusted operation environment when they use the system. Furthermore, traceability between the design and the implementation phase is achieved by verifying consistency and completeness between the architectures and the implemented system.

3.3.7. Verification phase

During the verification phase, we perform tests and reviews based on the implemented system. Tests include static/dynamic analysis and acceptance tests. In addition, since the vehicle is an integrated system based on many subsystems, an integration test is performed to determine whether there is a problem in the integrating procedure. At this point, integration means not only subsystem-based integration but includes hardware and software-based integration into the system. Subsequently, the penetration test is performed to determine whether a security threat exists for the integrated system. Also, minimum quality levels or documents are reviewed based on whether any design change made in the implementation phase does cause security, safety, and privacy-related problems.

3.3.8. Production & Release phase

The production and release phase produces the system and establishes response plans for possible accidents of the system after deployment. It is necessary to ensure the security of the vehicle production procedure since the vehicle has a long and complicated production procedure. This phase also reviews whether security issues occur in the deploying procedure of a produced system.

3.3.9. Operation phase

The operation phase monitors potential vulnerabilities of the system after deployment and responds to what is found. It is necessary to establish a monitoring plan and continuously monitor the system based on it. The operation phase also collects vulnerability reports on accidents and evaluates them to derive countermeasures. Then, corresponding countermeasures are disclosed and distributed with patches or updates. In the case of vehicles, it is important to have a team in charge of responding 24 hours a day since problems can be directly led to a casualty accident. Also, if a change occurs in a system in operation, the traceability of the system should be achieved by managing the configuration of the changes.

3.3.10. Disposal phase

In the disposal phase, the use of the system is terminated and the transfer or disposal procedure is performed. System transfer is the case for transferring owner and system disposal is the case for discarding vehicles. Although automobile manufacturers do not carry out the procedure of transfer or disposal on their own, they should ensure the trustworthiness of them by providing information to relevant partners with guidelines. The applicable guidelines should include information about the complete sanitation of the user's personal information and preservation of future available internal information (e.g. the mileage of the vehicle). Depending on the purpose, vehicles could be considered about various situations such as transferring owners (e.g. sales of used vehicles) or used by a large number of users (e.g. rental cars). Therefore, in the disposal phase, the automobile manufacturers must provide information to the responsible companies so that they can perform the procedure suitable for the use or characteristics of the system.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 November 2020 doi:10.20944/preprints202011.0644.v1

3.4. Detailed activities and evidences of each phase

This section presents detailed activities and evidences for the Trustworthy Automotive SDLC. After mapping the activities of Trustworthy Automotive SDLC with the contents of each evidence-based standard as shown in Figure 2, we derive detailed activities and evidences.

In this paper, we derived detailed activities by mapping 393 contents in the evidence-based standards (63 in CC, 104 in ISMS, 54 in PIMS, and 172 in FSMS) with 50 activities of our development process. As an example, the AGD class of CC contains the contents of determining whether a user can safely build an operating environment through the operation manual when distributing a product to a user. This can be mapped to the 6.1.2 guideline & tools creation of Trustworthy Automotive SDLC, through which the contents required by the AGD class user operation manual (e.g. The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.) can be reflected in this activity. As a result, the detailed activities of Trustworthy Automotive SDLC in terms of security, safety, and privacy are derived through this procedure. In addition, evidences (i.e. documents or templates) can be produced for each phase of Trustworthy Automotive SDLC in this procedure, and the list of evidences is shown in Table 3.

First, in the training phase, a training plan that includes the purpose of basic and advanced training, target audience, and topics covered in each training should be produced. Also, a list of attendance at training should be documented so that we can track the attendance or training list for each employee.

In the initiation phase, a project plan should be produced which includes the assets for each domain, roles (security team, development team, design team, etc.), tools, and the minimum quality level. In particular, the minimum quality level should be included as it serves as a baseline for whether or not to carry out the phase of the project. Also, project goals and the results of verifying consistency and completeness between goals in terms of security, safety, and privacy should be documented in a project goal verification report.

In the requirements analysis phase, an impact assessment report that evaluates the impact of assets by the domain (such as ASIL in terms of safety) and a requirements definition report based on this should be produced. Especially, the impact assessment report should include the results of the impact assessment on the reused existing system. In addition, it is necessary to derive a project requirements verification report that includes the analysis results, such as whether the requirements are consistently and completely derived based on the project goals or whether there is a conflict between the requirements of each domain.

In the acquisition phase, an acquisition plan including acquisition requirements and procedure for components developed by third-party should be produced. Based on this document, an acquisition inspection report including the results of evaluation and test performed independently on those components provided by other companies also should be derived.

In the design phase, a design specification for a unit (such as a module or interface) and an architectural specification for the entire system including all of them should be derived. Also, a system risk analysis report including mitigations should be produced based on the risk analysis results. This report should include the analysis result of the suitability of the mitigation for each domain and whether there is a conflict. A project design verification report also should be produced based on the analysis results of the consistency and completeness of the requirements.

In the implementation phase, a system implementation report including the source code based on the coding guidelines should be produced, and the user manual or tools including information related to the installation environment should be derived. At this phase, we also document the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 November 2020 doi:10.20944/preprints202011.0644.v1

consistency and completeness analysis result of the implementation performed based on the architecture in the project implementation verification report.

In the verification phase, we should produce a test result report including static and dynamic analysis results. In the case of an independent penetration test, the result should be documented in a vulnerability analysis document, and the final review report should be generated by reviewing whether all phases before production were performed properly.

In the production and release phase, a production plan including the plan and procedure of the product production should be derived, and the production verification report for the actual product should be documented. In addition, an incident response plan should be produced that includes a response plan for possible incidents in the future.

In the operation phase, a system monitoring report should be derived based on monitoring performed on servers or firewalls. In addition, an incident response report including the evaluation results of the discovered vulnerabilities, the disclosure of the vulnerabilities and the patch/update management plan should be produced.

Lastly, in the disposal phase, guidelines for system transfer and disposal should be derived for users and related companies so that the system can be securely transferred or disposed of even in situations that are not managed by the automobile manufacturer.

Table 3. Evidences of Trustworthy Automotive SDLC.

	Phase	Evidence
1	Training	· Training plan · List of attendance at training
2	Initiation	· Project plan · Project goal verification report
3	Requirements analysis	· Impact assessment report · Requirements definition report · Project requirements verification report
4	Acquisition	· Acquisition plan · Acquisition inspection report
5	Design	· Design specification · Architecture specification · System Risk Analysis report · Project design verification report
6	Implementation	· System implementation report · User manual or tool · Project implementation verification report
7	Verification	· Test result report · Vulnerability Analysis report · Final review report
8	Production & Release	· Production plan · Production verification report · Incident response plan
9	Operation	· System monitoring report · Incident response report
10	Disposal	· Guidelines for system transfer and disposal

4. Comparison and analysis of the research

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 November 2020 doi:10.20944/preprints202011.0644.v1

4.1. Comparison with secure SDLC standards and relevant studies

Trustworthy Automotive SDLC(TA_SDLC) is a security-by-design methodology which presents more detailed activities than the existing secure SDLC standards, and more suitable for automotive development. Table 4 is the result of a comparative analysis of existing secure SDLC standards and relevant studies by each requirement of Trustworthy Automotive SDLC. Microsoft SDL, NIST SSDLC, OWASP CLASP, and SAE J3061 were selected as the target secure SDLC standards, and studies targeting the entire development process from Section 2.1 were selected as the relevant studies [18], [19], [22], [29], [30]. The result has shown that secure SDLC standards and relevant studies were insufficient in presenting the security-by-design methodology suitable for automotive development. Notably, we found that all comparison targets were not satisfied with the 4th and 6th requirement. Existing secure SDLC standards have 36 activities with the depth of 2 throughout 9 phases in maximum, but Trustworthy Automotive SDLC consists of 50 activities with the depth of 2 throughout 10 phases. Therefore, we can say that the activities of Trustworthy Automotive SDLC have been sufficiently specified against the existing secure SDLC standards.

Table 4. Limitations of Secure SDLC standards and existing studies

Requir-ements	Secure SDLC standards				Existing studies					TA_SDLC
	Microsoft SDL	NIST SSDLC	OWASP CLASP	SAE J3061	[18]	[19]	[22]	[29]	[30]	
1	X	O	X	O	O	O	O	O	O	O
2	X	O	O	O	X	X	X	X	X	O
3	X	X	X	O	O	O	O	O	O	O
4	X	X	X	X	X	X	X	X	X	O
5	O	O	X	O	O	X	O	X	X	O
6	8/34	9/36	9/21	9/26	6/13	0/0	7/28	2/8	0/0	10/50

4.2. Analysis of UNECE regulation requirements

Our Trustworthy Automotive SDLC provides a development process suitable for automobile manufacturers targeting domestic as well as overseas markets by enabling them to respond to the upcoming UNECE regulation. To verify this, we chose 16 requirements related to the development process in UNECE regulation and mapped them to the activities of Trustworthy Automotive SDLC to determine whether they were satisfied. As a result, as shown in Table 5, it was confirmed that all of the requirements suggested by UNECE regulation could be satisfied with Trustworthy Automotive SDLC. Therefore, it can be said that the automotive security-by-design methodology we proposed has a form that can be certified against UNECE regulation.

5. Case study

In this chapter, we conducted a case study with a domestic automobile manufacturer A to prove the effectiveness of our methodology. Company A is an automobile manufacturer targeting not only domestic but also overseas markets and preparing to respond to UNECE regulation. We grasped the current development process and activities of company A based on interviews and surveys and analyzed to what extent the process satisfies the activities of Trustworthy Automotive SDLC.

As a result of mapping the development process of company A to Trustworthy Automotive SDLC, we found that 34 out of 50 activities suggested by Trustworthy Automotive SDLC were performed by company A as shown in Table 6. Therefore, it was derived that in order to achieve security-by-design, company A should perform 16 additional activities in the existing development process.

Table 5. Mapping between UNECE regulation requirements and TA_SDLC activities.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 November 2020 doi:10.20944/preprints202011.0644.v1

	UNECE regulation requirements (7.2. Requirements for the CSMS (Cyber Security Management System))	TA_SDLC activities (Activity number)
1	The vehicle manufacturer shall have a CSMS in place and shall comply with this Regulation.	Total
2	The vehicle manufacturer shall demonstrate that their CSMS applies to the development phase.	Total
3	CSMS shall include the processes used within the manufacturer's organization to manage cyber security.	Total
4	CSMS shall include the processes used for the identification of risks to vehicles.	3.1.1/3.1.2/5.2.1/ 5.2.3
5	CSMS shall include the processes used for the assessment, categorization and treatment of the risks identified.	5.2.2/5.2.3
6	The vehicle manufacturer shall demonstrate how their CSMS will manage dependencies that may exist with contracted suppliers, service providers or manufacturer's sub-organizations.	4.1.1/4.1.2/4.2.1
7	CSMS ensure security shall include the processes used for testing the cyber security of a vehicle.	7.1.1/7.1.2/7.1.3/7.1.4
8	CSMS ensure security shall include the processes in place to verify that the risks identified are appropriately managed.	7.2.1/9.1.1/9.1.2/9.3.1
9	The vehicle manufacturer shall demonstrate that their CSMS applies to the production phase.	8.1.1/8.1.2/8.1.3
10	The vehicle manufacturer shall demonstrate that the processes that include the capability to analyze and detect cyber threats, vulnerabilities and cyber-attacks from vehicle data and vehicle logs.	9.1.1/9.1.2/9.2.1
11	CSMS shall include the processes used for ensuring that the risk assessment is kept current.	9.2.4/9.3.1
12	CSMS shall include the processes used to monitor for, detect and respond to cyber-attacks, cyber threats, and vulnerabilities on vehicles.	9.1.1/9.1.2/9.2.1/9.2.2/9.2.3/ 9.2.4/9.3.1
13	CSMS shall include the processes used to assess whether the cyber security measures implemented are still effective in the light of new cyber threats and vulnerabilities that have been identified.	7.2.1/9.2.3
14	The vehicle manufacturer shall demonstrate that the processes used within their CSMS will ensure that cyber threats and vulnerabilities which require a response from the vehicle manufacturer shall be mitigated within a reasonable timeframe.	9.2.1/9.2.2/9.2.3/9.2.4
15	The vehicle manufacturer shall demonstrate that the processes that include vehicles after first registration in the monitoring.	9.1.1/9.1.2
16	The vehicle manufacturer shall demonstrate that their CSMS applies to the post-production phase.	8.1.1/8.1.2/8.1.3/8.2.1/8.3.1/ 9.1.1/9.1.2/9.2.1/9.2.2/9.2.3/9 .2.4/9.3.1/10.1.1/10.2.1

This is a result of applying Trustworthy Automotive SDLC to actual automobile manufacturers. We presented the possibility that based on our methodology, automobile manufacturers can improve the development process of their companies to satisfy UNECE regulation and to develop automobiles with reliability at the same time.

Table 6. Trustworthy Automotive SDLC activities fulfillment.

	Phase	Number of activities (A/TA_SDLC)
1	Training	2/3
2	Initiation	8/9
3	Requirements analysis	3/6
4	Acquisition	3/3
5	Design	5/6
6	Implementation	2/3
7	Verification	3/6
8	Production & Release	4/5

 doi:10.20944/preprints202011.0644.v1

9	Operation	4/7
10	Disposal	0/2
	Total	34/50

6. Conclusion

Traditional automotive development has focused on functional correctness and safety but not addressed security with emphasis. However, as the number of automotive hacking cases increase due to the recent increase in internet connectivity of vehicles, various international organizations are preparing cybersecurity regulations to achieve the security of automotive development. Typically, UNECE regulation will be applied on a new vehicle from 2022 and it emphasizes security-by-design which takes into account trustworthiness from the beginning of development. However, it does not provide a specific methodology for achieving security-by-design, and this is also true for previously researched studies. Therefore, to solve this problem, this paper proposes Trustworthy Automotive SDLC, a concrete methodology for automotive security-by-design.

In this paper, we first derived activities related to automotive development from 4 major secure SDLC standards and detailed them with 4 evidence-based standards CC, ISMS, PIMS, and FSMS in terms of security, privacy, and functional safety for the product and development environment. Additionally, based on the mapped results, we configured a detailed Trustworthy Automotive SDLC suitable for automotive development. We also demonstrated the effectiveness of applying Trustworthy Automotive SDLC by case study. Trustworthy Automotive SDLC considers all aspects of functional correctness, safety, and security which are important for automotive development. Furthermore, by consisting of activities embodied through existing secure SDLC standards and evidence-based standards, it can be used for the upcoming UNECE regulation. In future work, we will apply our methodology to the actual field and prove the effectiveness of it with the result.

Acknowledgments: This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2020-2015-0-00403)supervised by the IITP(Institute for Information &communications Technology Planning &Evaluation)

References

1. Bell, Ron. "Introduction to IEC 61508." ACM International Conference Proceeding Series. Vol. 162. 2006.
2. Barr, Michael. "Bookout vs. Toyota." case No. CJ-2008-7969, District Court of Oklahoma County, http://www. safetyresearch. net /Library/ Bookout_v_Toyota_Barr_redacted. pdf, consultado el 10 (2013).
3. Debouk, Rami. "Overview of the 2nd Edition of ISO 26262: Functional Safety–Road Vehicles." General Motors Company, Warren, MI, USA (2018).
4. Craigen, Dan, Nadia Diakun-Thibault, and Randy Purse. "Defining cybersecurity." Technology Innovation Management Review 4.10 (2014).
5. Mössinger, Jürgen. "Software in automotive systems." IEEE software 27.2 (2010): 92-94.
6. Miller, Charlie, and Chris Valasek. "Remote exploitation of an unaltered passenger vehicle." Black Hat USA 2015 (2015): 91.
7. M. Dehm, "Road Vehicles' Life-Cycle : Mapping of Relevant Standards and Regulations for Automotive Cybersecurity," Europe, 2019.
8. Khattri, Hareesh, Narasimha Kumar V. Mangipudi, and Salvador Mandujano. "Hsdl: A security development lifecycle for hardware technologies." 2012 IEEE International Symposium on Hardware-Oriented Security and Trust. IEEE, 2012.
9. Salini, P., and S. Kanmani. "Survey and analysis on security requirements engineering." Computers & Electrical Engineering 38.6 (2012): 1785-1797.
10. Khou, Stephen, et al. "A customizable framework for prioritizing systems security engineering processes, activities, and tasks." IEEE Access 5 (2017): 12878-12894.
11. Mohammed, Nabil M., et al. "Exploring software security approaches in software development lifecycle: A systematic mapping study." Computer Standards & Interfaces 50 (2017): 107-115.
12. Loruenser, Thomas, et al. "CryptSDLC: Embedding cryptographic engineering into secure software development lifecycle." Proceedings of the 13th International Conference on Availability, Reliability and Security. 2018.
13. Ruggieri, Maxwell, Tzu-Tang Hsu, and Md Liakat Ali. "Security Considerations for the Development of Secure Software Systems." 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON). IEEE, 2019.
14. Casola, Valentina, et al. "A novel Security-by-Design methodology: Modeling and assessing security by SLAs with a quantitative approach." Journal of Systems and Software 163 (2020): 110537.
15. Venson, Elaine, et al. "Costing secure software development: A systematic mapping study." Proceedings of the 14th International Conference on Availability, Reliability and Security. 2019.
16. Avizienis, Algirdas, et al. "Basic concepts and taxonomy of dependable and secure computing." IEEE transactions on dependable and secure computing 1.1 (2004): 11-33.
17. Michailidis, Alexander, et al. "Test front loading in early stages of automotive software development based on AUTOSAR." 2010 Design, Automation & Test in Europe Conference & Exhibition (DATE 2010). IEEE, 2010.
18. Takahira, Ricardo Y., et al. "Scrum and Embedded Software development for the automotive industry." Proceedings of PICMET'14 Conference: Portland International Center for Management of Engineering and Technology; Infrastructure and Service Integration. IEEE, 2014.
19. Young, William, and Nancy G. Leveson. "An integrated approach to safety and security based on systems theory." Communications of the ACM 57.2 (2014): 31-35.
20. Kriaa, Siwar, et al. "A survey of approaches combining safety and security for industrial control systems." Reliability engineering & system safety 139 (2015): 156-178.
21. Wolff, Carsten, et al. "AMALTHEA—Tailoring tools to projects in automotive software development." 2015 IEEE 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS). Vol. 2. IEEE, 2015.
22. Schmittner, Christoph, Zhendong Ma, and Erwin Schoitsch. "Combined safety and security development lifecylce." 2015 IEEE 13th International Conference on Industrial Informatics (INDIN). IEEE, 2015.
23. Sabaliauskaite, Giedre, Sridhar Adepu, and Aditya Mathur. "A six-step model for safety and security analysis of cyber-physical systems." International Conference on Critical Information Infrastructures Security. Springer, Cham, 2016.

24. Pricop, Emil, Sanda Florentina Mihalache, and Jaouhar Fattahi. "Innovative fuzzy approach on analyzing industrial control systems security." Recent Advances in Systems Safety and Security. Springer, Cham, 2016. 223-239.
25. Brunner, Michael, et al. "Towards an integrated model for safety and security requirements of cyber-physical systems." 2017 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C). IEEE, 2017.
26. Zhang, Yanan, et al. "Test and Evaluation System for Automotive Cybersecurity." 2018 IEEE International Conference on Computational Science and Engineering (CSE). IEEE, 2018.
27. Abdo, H., et al. "A safety/security risk analysis approach of Industrial Control Systems: A cyber bowtie–combining new version of attack tree with bowtie analysis." Computers & Security 72 (2018): 175-195.
28. Yi, Shengwei, et al. "A safety-security assessment approach for communication-based train control (cbtc) systems based on the extended fault tree." 2018 27th International Conference on Computer Communication and Networks (ICCCN). IEEE, 2018.
29. Skoglund, Martin, Fredrik Warg, and Behrooz Sangchoolie. "In search of synergies in a multi-concern development lifecycle: Safety and cybersecurity." International Conference on Computer Safety, Reliability, and Security. Springer, Cham, 2018.
30. Koschuch, Manuel, et al. "Safety & Security in the Context of Autonomous Driving." 2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE). IEEE, 2019.
31. Chowdhury, Thomas, et al. "Safe and secure automotive over-the-air updates." International Conference on Computer Safety, Reliability, and Security. Springer, Cham, 2018.
32. Asplund, Fredrik, et al. "Rapid Integration of CPS Security and Safety." IEEE Embedded Systems Letters 11.4 (2018): 111-114.
33. Lisova, Elena, Irfan Šljivo, and Aida Čaušević. "Safety and security co-analyses: A systematic literature review." IEEE Systems Journal 13.3 (2018): 2189-2200.
34. Geismann, Johannes, Christopher Gerking, and Eric Bodden. "Towards ensuring security-by-design in cyber-physical systems engineering processes." Proceedings of the 2018 International Conference on Software and System Process. 2018.
35. Huang, Kaixing, et al. "Assessing the physical impact of cyberattacks on industrial cyber-physical systems." IEEE Transactions on Industrial Electronics 65.10 (2018): 8153-8162.
36. Fowler, Daniel S., et al. "A Method for Constructing Automotive Cybersecurity Tests, a CAN Fuzz Testing Example." 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C). IEEE, 2019.
37. Oka, Dennis Kengo, Tommi Makila, and Rikke Kuipers. "Integrating Application Security Testing Tools into ALM Tools in the Automotive Industry." 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C). IEEE, 2019.
38. Verma, Siddhartha, et al. "Combined approach for safety and security." International Conference on Computer Safety, Reliability, and Security. Springer, Cham, 2019.
39. Apvrille, Ludovic, and Letitia W. Li. "Harmonizing safety, security and performance requirements in embedded systems." 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE). IEEE, 2019.
40. Dobaj, Jürgen, et al. "Towards Integrated Quantitative Security and Safety Risk Assessment." International Conference on Computer Safety, Reliability, and Security. Springer, Cham, 2019.
41. Bramberger, Robert, et al. "Co-engineering of Safety and Security Life Cycles for Engineering of Automotive Systems." ACM SIGAda Ada Letters 39.2 (2020): 41-48.
42. De Win, Bart, et al. "On the secure software development process: CLASP, SDL and Touchpoints compared." Information and software technology 51.7 (2009): 1152-1171.
43. Microsoft, "Security Development Lifecycle - SDL Process Guidance Version 5.2," 2012.
44. R. Kissel et al., "Sp 800-64 rev. 2. security considerations in the system development life cycle," 2008.
45. OWASP, "Comprehensive, lightweight application security process," 2006. [Online]. Available: http://www.owasp.org
46. SAE Vehicle Electrical System Security Committee, "Sae j3061-cybersecurity guidebook for cyber-physical automotive systems,", 2016.
47. Schmittner, Christoph, and Georg Macher. "Automotive Cybersecurity Standards-Relation and Overview." International Conference on Computer Safety, Reliability, and Security. Springer, Cham, 2019.
48. UNECE, "Draft Cyber Security Regulation - final clean version," 2020.

49. Hunjan, H. "ISO/SAE 21434 Automotive Cyber-Security Engineering." Presentation, Renessas Electronics LTD (2018).
50. Schmittner, Christoph, Gerhard Griessnig, and Zhendong Ma. "Status of the Development of ISO/SAE 21434." European Conference on Software Process Improvement. Springer, Cham, 2018.
51. Blyler, John. Software-Hardware Integration in Automotive Product Development. SAE, 2014.
52. LDRA, "BUILD SECURITY INTO THE CONNECTED CAR DEVELOPMENT LIFE CYCLE," 2017.
53. Schoitsch, Erwin, et al. "The need for safety and cyber-security co-engineering and standardization for highly automated automotive vehicles." Advanced Microsystems for Automotive Applications 2015. Springer, Cham, 2016. 251-261.
54. Sabaliauskaite, Giedre, and Aditya P. Mathur. "Aligning cyber-physical system safety and security." Complex Systems Design & Management Asia. Springer, Cham, 2015. 41-53.
55. Synopsys, "What is ASIL?"
56. Schmittner, Christoph, and Zhendong Ma. "Towards a framework for alignment between automotive safety and security standards." International Conference on Computer Safety, Reliability, and Security. Springer, Cham, 2014.
57. Miller, Joseph D. Automotive System Safety: Critical Considerations for Engineering and Effective Management. John Wiley & Sons, 2019.
58. Mellado, Daniel, Eduardo Fernández-Medina, and Mario Piattini. "A common criteria based security requirements engineering process for the development of secure information systems." Computer standards & interfaces 29.2 (2007): 244-253.
59. Yin, Lei, and Fang-Liang Qiu. "A novel method of security requirements development integrated common criteria." 2010 International Conference On Computer Design and Applications. Vol. 5. IEEE, 2010.
60. Mellado, Daniel, et al. "A systematic review of security requirements engineering." Computer Standards & Interfaces 32.4 (2010): 153-165.
61. Houmb, Siv Hilde, et al. "Eliciting security requirements and tracing them to design: an integration of Common Criteria, heuristics, and UMLsec." Requirements Engineering 15.1 (2010): 63-93.
62. Mesquida, Antoni Lluís, and Antonia Mas. "Implementing information security best practices on software lifecycle processes: The ISO/IEC 15504 Security Extension." Computers & Security 48 (2015): 19-34.
63. Li, Hongbo, et al. "Fesr: A framework for eliciting security requirements based on integration of common criteria and weakness detection formal model." 2017 IEEE International Conference on Software Quality, Reliability and Security (QRS). IEEE, 2017.
64. Barafort, Béatrix, Antoni-Lluís Mesquida, and Antonia Mas. "Integrating risk management in IT settings from ISO standards and management systems perspectives." Computer Standards & Interfaces 54 (2017): 176-185.
65. Barafort, Béatrix, Antoni-Lluís Mesquida, and Antònia Mas. "Integrated risk management process assessment model for IT organizations based on ISO 31000 in an ISO multi-standards context." Computer Standards & Interfaces 60 (2018): 57-66.
66. Lee, Younghwa, Jintae Lee, and Zoonky Lee. "Integrating software lifecycle process standards with security engineering." Computers & Security 21.4 (2002): 345-355.
67. Horie, Daisuke, et al. "A new model of software life cycle processes for consistent design, development, management, and maintenance of secure information systems." 2009 Eighth IEEE/ACIS International Conference on Computer and Information Science. IEEE, 2009.
68. Amara, Naseer, Zhiqui Huang, and Awais Ali. "Modelling Security Requirements for Software Development with Common Criteria." International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage. Springer, Cham, 2019.
69. Hatton, Les. "Safer language subsets: an overview and a case history, MISRA C." Information and Software Technology 46.7 (2004): 465-472.