Joint functional safety ISO 26262 and cybersecurity STRIDE/HEAVENS assessment by developers within MBSE SPES framework using extended SysML diagrams and minor automations

Ivo Häring [a], Vivek Sudheendran [b], Roman Sankin [c], Stefan Hiermaier [d]

[a] Fraunhofer Institute for High-Speed Dynamics, Ernst-Mach-Institut, EMI, Efringen-Kirchen, Germany, ivo.haering@emi.fraunhofer.de,

[b] Work done at Bosch Engineering GmbH, Current Affiliation: Deutsches Elektronen-Synchrotron DESY, Hamburg, Germany, vivek.sudheendran@desy.de

[c] Bosch Engineering GmbH, Abstatt, Germany, roman.sankin@de.bosch.com

[d] Department of Sustainable Systems Engineering, INATECH, University of Freiburg, Germany, stefan.hiermaier@inatech.uni-freiburg.de

Abstract: To manage the increasing complexity of modern automotive systems, development companies adhere to model based systems engineering (MBSE). Within MBSE processes, suitable modeling approaches need to be selected and combined. Modelling and simulation approaches include semi-formal modeling, software generation, engineering simulation and software emulation. Today, even the selection, tailoring and interfacing of modeling approaches can be supported within framing methodologies. Within such a digitalized development process context, the paper addresses the question how to use SysML modeling to support efficiently the functional safety as well as the cybersecurity (IT security) assessment of developers within the early stages of the system development process in the automotive domain within MBSE. The feasibility of the approach is realized by the development of a concept for functional safety and cybersecurity analysis within the Software Platform Embedded Systems (SPES) framework. The concept is documented with metamodels and is backed by SysML profiles which extend the SPES profile within the IBM Rational Rhapsody environment. The profile for cybersecurity analysis supports assessment of developers at the system level adhering to the guidelines of the Microsoft STRIDE based HEAling Vulnerabilities to Enhance Software Security and Safety (HEAVENS) security model, specifically for automotive. SysML model-based prototypes, i.e. SysML system designs including their functional safety and cybersecurity assessment, are developed, which validate the approach within an automotive MBSE pilot project. A sample prototype application shows the feasibility of the approach and allows to estimate the effort of SysML supported functional safety and cybersecurity assessments of developers within a SPES conform environment. Main results include the feasibility of reuse and further development of SPES oriented SysML models (e.g. context, scenario, goal, function) intended for system design. The functional safety and cybersecurity relevant model extensions and refinements are realized within these system models. The refinements and extensions result in functional safety relevant models which support item definition, hazard and risk analysis, functional safety concept and technical safety concept. Similarly, cybersecurity relevant SysML models help in Target of Evaluation (TOE) description, threat analysis and risk assessment and cybersecurity requirement derivation according to the HEAVENS approach. The automations imparted on these extended SysML models by using helpers enhance the usability. For instance, helpers provide automatic functional safety and cybersecurity parameter determination within models (e.g. ASIL determination, security level derivation) and filtered graphical views based on inputs of developers. Together with a model checker they assist fast execution of the analyses, consistency checks and generation of the assessment artifacts, e.g. tabular overview of risks and their control.

1. INTRODUCTION

The deployment of the systems engineering approach in the automotive domain is enforced by growing complexity, integration needs, complex supply chains [1] and increasing (functional) safety and security demands [2]. Model based systems engineering (MBSE) [3] applies systems modeling as part of the

systems engineering process to support analysis, specification, design, tests, verification and validation of the system being developed, e.g. as required by the V-model [4]. The important artifact of such an approach is the development of a coherent shared digital model of the system. This has many advantages compared to (digital) document-based approach by improving the specification, enhancing the design quality and consistency, reusability of design and specification artefacts and better communication among the development teams. Thus the quality aspects such as consistency, understandability and well-formedness are enhanced [3].

The systems modelling language (SysML) [5] is a general-purpose semi-formal graphical language for modeling systems that consist of hardware, software, data and other elements within a physical-engineered environment. It is used for the modeling of requirements, of structure and behavior, and related parameters to provide a description of a system, its components and its surroundings. Thus, SysML is a good candidate to implement an MBSE approach. The effective use of SysML is possible only if a well-structured methodology is followed [6]. In addition, its way of application needs to be adopted to the MBSE development process. In particular, to go beyond proof of principle applications, e.g. to functional safety [7] [8] [9] [10].

The frequent failure of current model-based development approaches in practice is mainly due to lack of modeling (of modeling) theories and missing integration of theories, methods and tools. Another issue is that tool chains are tailored according to existing development processes rather than development processes adopted and updated with respect to best practice available tools, in particular in big companies [11]. The transition from one model to another is error prone. This is because models applied are often based on separated and unrelated modeling theories [12].

The competitive pressure is very high in the embedded systems market, thus time to market and development costs are strong drivers. The inability to deliver high quality embedded systems can cause direct financial consequences or threat to people [13]. For instance, the well-known vehicle recalls from Ford [14] and Toyota [15]. Other examples include the infamous delivery delays of the IEC train generation 3 by Siemens due to the safety issues [16] or the more recent safety problems in Boeing [17]. In addition, the IOT (Internet of Things) approaches, connected infotainment, car communication in wireless networks, remote updating and maintenance, etc., pave the way to cybersecurity measures.

Many similarities are identified between safety and security processes. This overlap can be used to define a comparable process flow and assessment technique between both dependability topics [18]. However, when compared to (functional) safety processes, IT-security assessment and generation processes are less established [19], in particular in the (automotive) embedded domain, even less when aiming at joint or similar modeling approaches [20].

A typical scenario in automotive supply chains is that engineering service providers, e.g. [21], work for Original Equipment Manufacturers (OEMs) and offer services in the development of systems, functions and software for powertrain, safety, vehicle dynamic and infotainment system, and electrical and electronic integration of these systems, the complexity of which ask for the MBSE approach. In addition, the functional safety and cybersecurity requirements in the projects are crucial. Most relevant standards are ISO26262 [22] for automotive functional safety, Automotive SPICE (ASPICE) ISO/IEC 33004 [23] for process management, Safety of Intended Functionality (SOTIF) ISO/PAS 21448 [24] for autonomous driving, and the novel cybersecurity standard ISO/PAS 21434 [25], etc. However, such processes are now still heavily based on digital document-based approaches and often not conducted jointly.

In this scenario, the motivation of the paper is to help functional safety and cybersecurity engineers in their assessment work in a complex project environment in the early phases of system design. In particular, to support them in the preparation of SysML artefacts covering functional safety and cybersecurity risk assessments, requirements determination of functional safety and cyber security, also regarding related method selection in terms of assignment of levels of rigor for development. This is implemented on design concept level and for later verification and validation. Hence the present paper

addresses the following best practice research gaps: (i) How to integrate functional safety and cybersecurity analysis within the MBSE approaches like SPES? (ii) How to reduce the efforts of manual approaches to system development with respect to traceability, completeness, partial automation? (iii) How do model based approaches help for more effective management of complex development lifecycles when compared to document-based approaches? (iv) How to substantially support the implementation of best practice functional safety and cybersecurity standards in MBSE approaches using semiformal models? (v) How to validate the approach?

The paper is structured as follows. Section 2 shows the main research gaps regarding MBSE in the automotive domain for functional safety assessment and cybersecurity assessment using semiformal models. Section 3 describes analysis work done to map different processes in functional safety and cybersecurity assessment of the automotive domain to model based methodologies and semi-formal language paradigms. The concepts are documented as metamodels. It also explains further approach specifications and hints at criteria-based concept validation, respectively using the SysML metamodels. Section 4 deals with the implementation of the approach and its validation by prototyping. The stereotypes, tags used in various SysML diagrams, the automations realized using these models and the helpers and model checker applications are tested regarding the support of functional safety and cybersecurity assessment of developers. Finally, discussion and assessment at implementation level is conducted. Section 5 presents conclusion and outlook.

2. STATE-OF-THE-ART AND GAPS: SEMI-FORMAL BASED SAFETY AND SECURITY ASSESSMENT WITHIN MBSE USING SPES XT AND SYSML

Not focusing on a core-centric manner, i.e., implementation-relevant documentation, it is challenging to shift to a more effective process that is requirement-driven and architecture centric [26]. To this end The International Council on Systems Engineering (INCOSE) came up with the MBSE methodology which is based on ISO/IEC/IEEE 15288 standard [27]. Along these lines, the Software Platform for Embedded Systems (SPES) [12] is a model-based methodology which aims at seamless integration of different related processes and models. It focuses on the modeling of safety relevant aspects by integrating the safety aspects within the development processes and its models using analyses models, which are developed in accordance with the principles of the overall model-based development (MBD), aiming at better consistency and traceability among different safety analyses methods and techniques [28]. It helps in leveraging synergies between integration of different models [26]. The main concept used in SPES is the use of abstraction layers, views and viewpoints [29]. The system under development can be designed at various levels of abstraction based on their structural significance. Views and viewpoints are used to handle the concerns of different stakeholders in the complicated engineering process [29]. A view represents a whole system from the perspective of a related set of concerns and viewpoint which defines how to use the view [30].

When comparing IBM Telelogic Harmony-SE, INCOSE Object Oriented Systems Engineering Method (OOSEM), IBM Rational Unified Process for Systems Engineering (RUP SE), Vitech MBSE in extension of [31], SPES [32] is the only approach along with Vitech MBSE which covers the attributes framework, functional safety, SysML, simulation and integration of models. This shows the significance of the SPES approach. However, none of the listed MBSE frameworks covers cyber security. Other dimensions could be costs of the approach or tools and number of publications referring to the approach, i.e. scientific acceptance, user acceptance, level of formal proofing etc.

The research gaps that have not been addressed in SPES 2020 include the quality aspects, namely variability, early artifact validation, etc., which are prerequisites for the development process to be able to apply the SPES framework [32]. The extended version of SPES 2020 methodology called SPES XT [13] focused on solving some of the problems and challenges in SPES 2020. SPES XT improves the integration between software and systems engineering, optimal deployment of software to hardware, early validation and modular safety assurance. The modular safety assurance is based on the Open Safety Model [13]. Again, the cybersecurity aspects are also not discussed in SPES XT.

Functional safety has already been assessed using SysML models at various level of abstraction, see [33]. However, often not within a MBSE context, i.e. not being framed by an overall modelling environment such as SPES for the (automotive) embedded domain. Also, rather few applications focus on the concept phase of ISO 26262 regarding hazard and risk analysis (HARA), as in the present approach, but rather on using SysML for formalization visualization.

In general, the cybersecurity evaluation methodologies to identify security vulnerabilities in the automotive E/E systems are not well established as compared to safety [34]. The dedicated cybersecurity standard for the automotive domain is the rather new standard ISO/PAS 21434 [25]. The HEAVENS [18] (HEAling Vulnerabilities to Enhance Software Security and Safety) security model aims to provide a framework to identify security needs in complex embedded systems in the automotive domain. The model is developed mainly based on the state of the art of threat analysis and risk assessment methodologies used in other non-automotive industries like telecommunication, IT, defense, and web applications [34].

For informed selection, different cybersecurity methodologies and dependency with HEAVENS are compared in extension of [34] considering application domain, coverage of threat, vulnerability and risk assessment: Common Criteria (CC), Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE), The European Telecommunications Standards Institute (ETSI), Open Web Application Security Project (OWASP), Microsoft STRIDE [35], E-safety Vehicle Intrusion Protected Applications (EVITA), SECTRA model, Common Vulnerability Scoring System (CVSS), and Common Vulnerability Scoring System (CVSS). For threat analysis, STRIDE is selected because it categorizes the threats in a generic way and is applicable to the embedded automotive domain.

For instance, the threats which are specific as described in EVITA [36] come as subset of the threat categories in STRIDE. Moreover STRIDE is threat centric, which means that final outputs of attacks get highlighted rather than focusing on specific attacks [35]. The vulnerabilities are exploited by threats from attackers. This causes risks to the assets. An injective mapping of threat to vulnerability is not feasible since a vulnerability can lead to many threats and a single threat can exploit many vulnerabilities. Common Criteria (CC) [37] and WIFFs [38] provide generic mappings only. The risk assessment part of HEAVENS derives the security level from the threat level, which represents the attack potential, and the impact level, which relates to the impact component of the risk.

Parameters used to calculate threat level in attributes in different methodologies can be compared for the just listed security assessment frameworks. The threat level parameters are mainly in alignment with CC. In HAEVENS, the elapsed time is not considered as compared to CC, because of its use of the following parameters (categories): expertise, knowledge about target of evaluation (TOE), window of opportunity, and equipment. Parameters used to calculate the impact level in different methodologies can as well be compared. The possible consequences of successful attacks on the TOE is determined considering the impact level factors. HEAVENS uses, similar to the business impact factors of OWASP and EVITA, the factors safety, financial, operational, privacy, and legislation.

In summary, also further reviewed works hint at the feasibility and usefulness of SysML/UML extensions for system cyber security assessment of developers also jointly with functional safety and as starting point for in parts automated assessments in the concept phase of automotive (embedded) system developments. However, they are so far typically applied only in slightly other fields such as networks, at organizational level, IoT-Applications, Industrial Control Systems (ICS) and focus on the application of specific methods and are also not embedded in a MBSE framework.

3 ANALYSIS, CONCEPT DEVELOPMENT AND METAMODELLING

For brevity of presentation and explicit illustration the focus is from now onwards on the SysML supported cyber-security assessment according to the HEAVENS approach within the SPES MBSE framework, for a similar approach for functional safety see [33]. HEAVENS's cybersecurity assessment

can be divided into three steps, which need to be covered and documented by the system developer, see Figure 1:

1. Threat Analysis: The TOE is the feature or subject for analysis. It is analyzed for STRIDE threats. The TOE or its dependent functional use cases are represented in Data Flow Diagrams (DFDs). The identified security threats are mapped to the assets. Assets are entities in a data flow diagram which are classified as process, data flow, data store or external entity. The mapping is done between threats and security attributes.
2. Risk Assessment: The mapping between STRIDE threats and assets, impact levels and threat levels are used to determine the security level.
3. Cybersecurity Requirement derivation: The mapping between STRIDE threats and assets and security levels are considered to formulate security requirements.

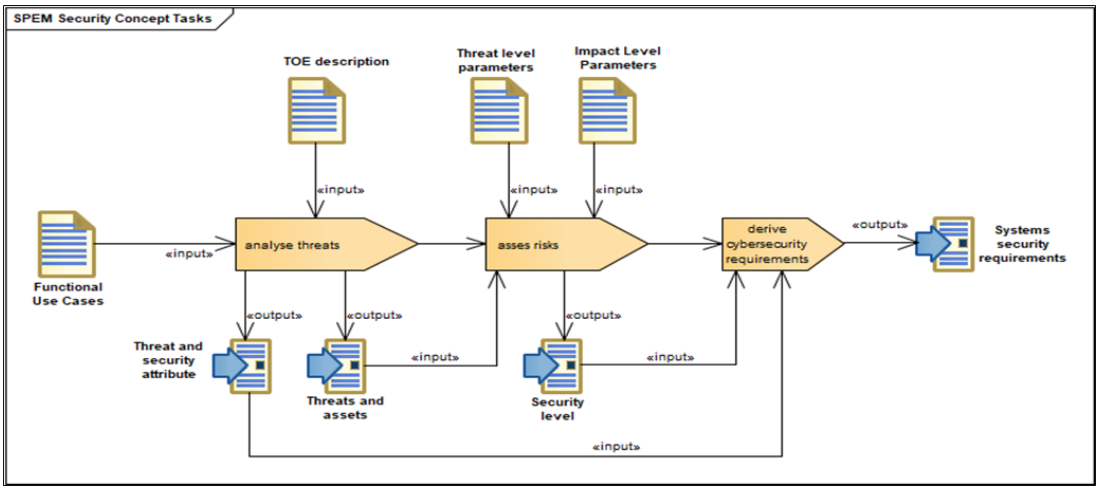

Figure 1: Work flow of HEAVENS [34] in OMG SPEM (Software and Systems Process Engineering Metamodel) diagram [39].

Table 1 shows the STRIDE threats with security attribute mapping, while also explaining the acronym itself [40] [34].

Table 1: STRIDE threat and security attribute mapping [34].

STRIDE Threat	Description	Security attribute
Spoofing (S)	attackers pretend to be someone or something else	Authenticity, Freshness
Tampering (T)	attackers change data in transit or in a data store, attackers may change functions as well – implemented in software, firmware or hardware	Integrity
Repudiation (R)	attackers perform actions that cannot be traced back to them	Non –repudiation, Freshness
Information disclosure (I)	attackers get access to data in transit or in a data store	Confidentiality, Privacy
Denial of service (D)	attackers interrupt a system's legitimate operation	Availability
Elevation of privilege (E)	attackers perform actions they are not authorized to perform	Authorization

In the STRIDE-per-element variant, every element in the DFD is evaluated for STRIDE threats. In STRIDE-per-interaction variant, the interaction of every element in the DFD is the basis for threat evaluation. For the automotive domain application, the STRIDE-per-element variant is better than the STRIDE-per-interaction [41]. Table 2 tabulates the STRIDE-per-element mapping with DFD elements.

Table 2: SRIDE-per-element mapping [41].

DFD element type	S	T	R	I	D	E
External entity	x		x			
Data Flow		x		x	x	
Data store		x	x	x	x	
Process	x	x	x	x	x	x

The three analysis steps of HEAVENS (see throughout also Figure 1) covered by the MBSE approach can be summarized, using Table 1 and 2:

- Cybersecurity assessment as per HEAVENS model uses Microsoft's STRIDE threat model [34]. The concept of TOE is the starting point for the assessment. It is similar to Item as used in ISO26262. Threat as used in STRIDE is the security counterpart in cybersecurity similar to hazard as used in ISO26262.
- The HEAVENS methodology performs threat analysis and risk assessment (TARA) to derive security levels, which are comparable to Automotive Safety Integrity Level (ASIL). The threat analysis is conducted based on data flow diagrams of the functional use cases related to TOEs. STRIDE threats are assigned to the assets for each TOE. Threats are mapped to security attributes which are also assigned to the asset.
- Impact level and threat are the 2 parameters used within the risk assessment to derive the security level. The cyber security requirement is assigned to an asset. The asset will be mapped to its associated threat and security attribute and also to the security level. The asset assigned with threat, security attribute and security level are the main attributes connected to a HEAVENS's cybersecurity requirement.

In the present application context, the system and software development consist of models in 3 layers. A higher layer which is context relevant. The context here represents the project as well as technical level. The next level defines the system and the third layer focuses on hardware (HW) and software (SW). These levels are mapped to the abstraction layers in SPES which are Context Layer, System Layer and HW/SW Layer. In each of these three abstraction layers, the viewpoints are developed namely, Requirement viewpoint, Functional viewpoint, Logical viewpoint and Technical viewpoint.

Apart from the mapping of abstraction layers and viewpoints with different models relevant to safety and security attributes, the concept development incorporates automation. The automation is realized using helpers [42] which are independent of abstraction layers and viewpoints. Helpers are custom programs that one attaches to the models to enhance their functionality. A model checker implemented with custom checks is an add-on to the approach generating additional artefacts based on the system model and the assessment inputs of the system developer. In general, this mapping structure can be used in all projects. But all the four viewpoints may not be needed for safety or security assessment in every layer. The SysML support in such processes enhances the efficiency of the work in the case of cyber security assessment. In Table 3, SPES mapping with HEAVENS phases are tabulated.

Table 3: SPES mapping with HEAVENS workflow phases.

	Viewpoints			
Abstraction layers	**Requirement**	**Functional**	**Logical**	**Technical**
Context	TOE description, Cybersecurity requirement derivation	TOE description, Threat analysis, Risk assessment	TOE description, Threat analysis, Risk assessment	
System	Cybersecurity requirement derivation	Threat analysis, Risk assessment	Threat analysis, Risk assessment	Threat analysis, Risk assessment
HW/SW	Cybersecurity requirement derivation	Threat analysis, Risk assessment	Threat analysis, Risk assessment	Threat analysis, Risk assessment

Next, SysML metamodels are used to represent the SPES methodology and also to represent the present extension of SPES to IT-security allowing its use by developers within MBSE environments like IBM Rhapsody. The graphical/semi-formal metamodeling makes the steps and dependencies more transparent.

Figure 2 and Figure 3 show the Block Definition Diagram (BDD) metamodels for cybersecurity assessment as applied by system developers. In Figure 2, the TOE description is shown. TOE shows similar associations with SysML model types as can be used in the case of Item. Similarly, the goal model according to ISO 26262 is replaced by a data flow model in case of cybersecurity assessment. DataFlowModel has aggregation including SysML Activity Diagrams (ADs), SysML Block Definition Diagrams (BDDs) [43] [44], etc. The arrows and lines show the relations among different entities that

that can be used by the developer. In [45], there are similar uses of diagram types for SysML metamodeling.

In Figure 3, the threat analysis and risk assessment metamodel is shown. Each TOE has composition (i.e., strong aggregation) of Asset. Each Asset has a number of STRIDEThreat tags of type STRIDE. The Asset is generalized by the application of stereotypes <<Process>>, <<DataFlow>>, <<DataFlowBlock>>, <<DataStore>>, and <<ExternalEntity>>. These stereotypes create a list of STRIDE threats within the asset. <<CyberSecurityRequirement>> is a generalization of SysML Requirement. It is allocated to the Asset.

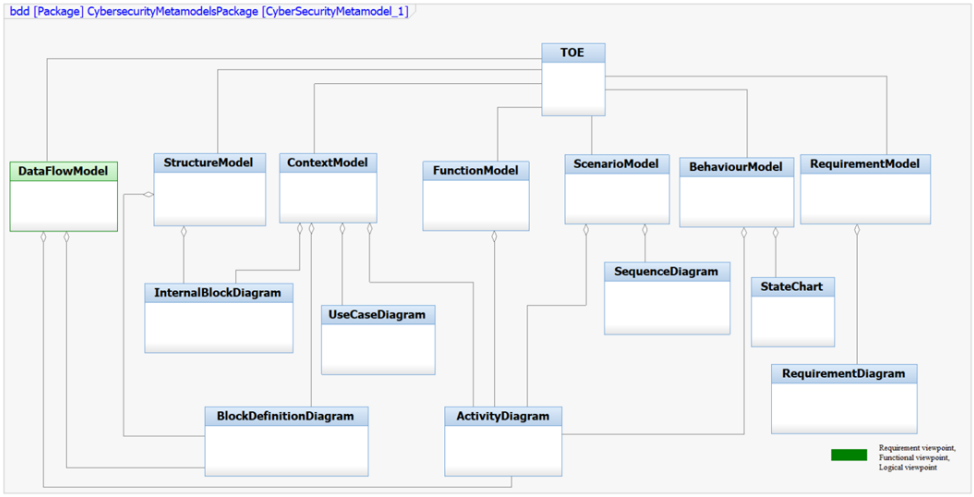

Figure 2: BDD metamodel for cybersecurity assessment showing TOE description that can be used by developer.

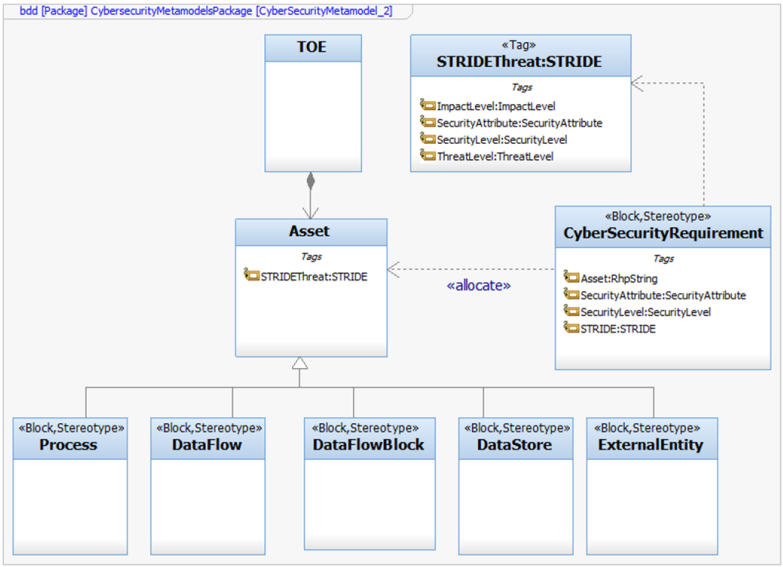

Figure 3: Metamodel for cybersecurity assessment of HEAVENS workflow for risk assessment.

The SysML metamodel can be assessed regarding the coverage of the HEAVENS cyber-security approach and its capability to relate to relevant SysML diagrams of the system models showing that TOE description is covered using BDD, Use Case Diagrams (UCD), ADs, and Internal Block definition Diagrams (IBD) and related associations [46, 47]. Threat analysis is covered by stereotypes that generate tags in corresponding model elements like block, object flow, actor, etc. Risk assessment is conducted using helpers, and cybersecurity requirements can be modeled using SysML Requirement Diagram (RD), also by extracting its information in a tabular form. Thus, completeness, traceability and automatability of the approach are ensured.

4. IMPLEMENTATION, PROTOTYPING, RESULTS AND DISCUSSION

The implementation and validation of the approach based on the metamodels is done by prototyping. The section contains SPES packaging associated with the concept, the related models, automation combined with stereotyped models and the discussion on model checking. It elaborates the validation of the concept with the help of prototypes in a pilot system engineering project. It also discusses a quantitative assessment of the implementation.

SPES abstraction layers and viewpoints (see section 2) are visualized in the SysML Package Diagram (PD) structure in Figure 4 to show the organization of SysML models in different packages, similar as in [5, 48]. Note that views and viewpoints are a collection of models viewed together. These models or a single model can spread among several abstraction layers.

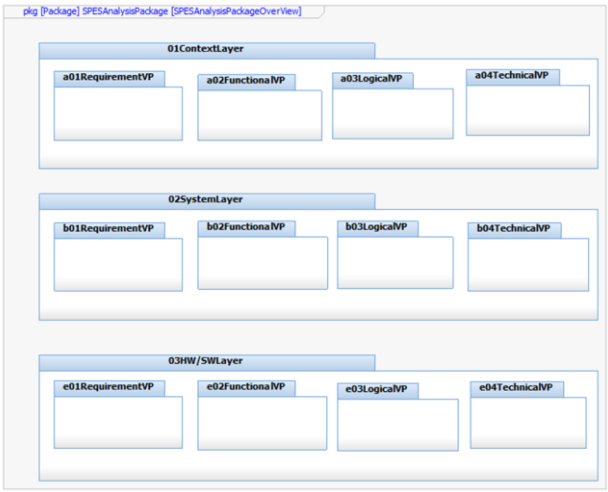

Figure 4: SPES oriented and extended package structure.

As introduced in section 3, the various models required to integrate SPES, HEAVENS, ISO26262 and SysML are context model, scenario model, goal model, requirement model, function model, structure model, behavior model and data flow model. Note that the first five are standard modeling names in SPES [49, 50], whereas the last three have been introduced specifically. Each model consists of suitable combinations of SysML diagram types as presented in section 3.

In Table 4, the mapping of model elements which relate with SPES, ISO26262 and HEAVENS are distributed among different SPES abstraction layers and viewpoints.

Table 4: Mapping of model elements used in SPES's abstraction layers and viewpoints.

	Viewpoints			
Abstraction Layers	**Requirement**	**Functional**	**Logical**	**Technical**
Context	Context, scenario, goal, requirement (functional safety and cybersecurity)	Item function, TOE function, functional safety concept	Item, hazard, operational situation, hazard event, TOE, asset, threat	
System	Requirement (functional safety and cybersecurity)	System function, technical safety concept, function associated with asset	Logical components in item (system), asset, threat	ECU, sensor, actuator, interface
HW/SW	Requirement (functional safety and cybersecurity)	SW function, safety related SW function, HW function	Logical SW, HW elements, asset, threat	HW and SW classes, interfaces

For automation and model checking, IBM Rational Rhapsody provides application peripheral interfaces (APIs) [51] in programming languages, C++ and Java. The helpers as introduced in section 3 invoke the programmed automation scripts that support SysML models [42]. In present approach, Java APIs for the Rhapsody model interface and Java interfaces for the model checker are used to implement automations using the Rhapsody software development kit (SDK) in Eclipse IDE environment.

The following automation concepts are used:

1. The security level is automatically determined from the impact level and threat level assigned to each STRIDE threat by the developer. The threat level and impact level are also calculated automatically based on its parameters. The security attribute related to the threat is also mapped to the threat.
2. For implementation of the approach, a third-party model checker has been extended and custom checks are using Java plugins. The third party model checker includes a custom user interface which displays the checks and their results.
3. Diagram views were implemented to graphically isolate certain modelling elements to improve readability and reduce complexity.

For instance, the color change is applied to the graphical elements in an activity diagram which shows the technical safety concept based on and visualizing assessment inputs. This helps to distinguish normal functions from safety relevant functions graphically. Note that this feature is different from diagram views.

The validation of the approach is performed using and extending available SysML model-based prototypes of a pilot project undertaken in Bosch Engineering GmbH. The system Malfunction Indicator Lamp (MIL) Heteronomy and its features are chosen for which the development and analysis are carried out in SysML in an IBM Rhapsody environment. It indicates the malfunction or fault in the engine management system of interest (SOI) via the instrument cluster of the vehicle. The term heteronomy is used because the MIL system indicates fault in engine system due to foreign or external factors affecting the engine. For instance, ESP (Electronic Stability Program) [52] break down can lead to higher engine emission. MIL heteronomy indicates this issue via the cluster.

The cybersecurity assessment approach is based on TOE description, see Figure 2. The assets are classified by the developer according to HEAVENS based on following stereotypes: <<Process>> applicable to SysML block and swimlane, <<DataStore>> applicable to SysML block and swimlane, <<DataFlow>> applicable to object flow in activity diagram and swimlane, <<ExternalEntity>> applied to actor in use case diagram, <<DataFlowBlock>> is an additional stereotype that serves the same purpose of <<DataFlow>> but is applied to the block instead of the object flow.

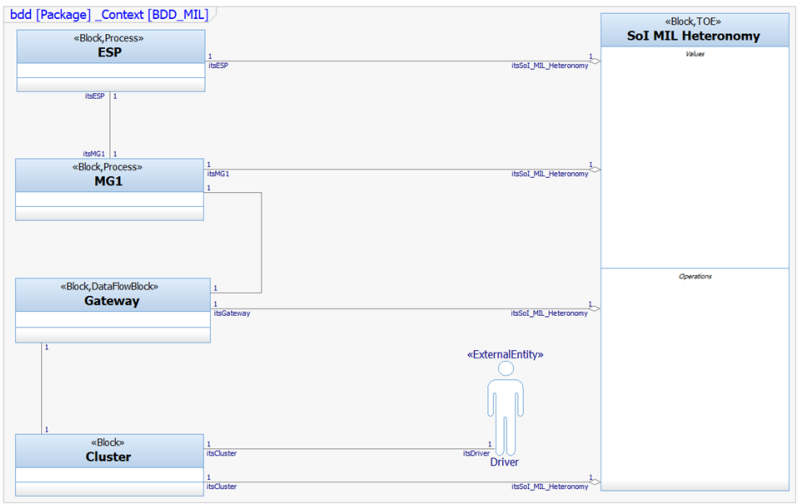

Figure 5: Cybersecurity stereotypes applied to block elements of TOE.

The STRIDE threats associated with the stereotyped assets are listed as tags. The threat analysis phase is initiated by the application of stereotypes to the diagram elements by the developer as shown in Figure 5. The stereotypes make the diagram components cybersecurity relevant and hence requiring further assessment by the developer. As an example, in Figure 5, to the block ESP the stereotype << Process>> is applied. This is due to the fact that ESP symbolizes a logical electronic control unit (ECU) system and this is a process element according to the data flow model based on HEAVENS. Similarly consider Gateway: it is related to the dataflow element in dataflow model in HEAVENS. Hence the stereotype <<DataFlowBlock>> is used. Such diagrams can also be used to color cybersecurity relevant elements.

These diagram elements are the Assets associated with TOE in the HEAVENS model. The stereotypes generate threats as SysML tags within the Assets. In Figure 6(a), the Asset ESP which is a process entity, is mapped to the list of STRIDE threats, see Table 1. The cybersecurity attribute mapped to each threat is generated as a tag within Threat. Similarly, all the elements with cybersecurity relevant stereotypes Process, DataFlow, ExternalEntity, DataStore and DataFlowBlock generate threats.

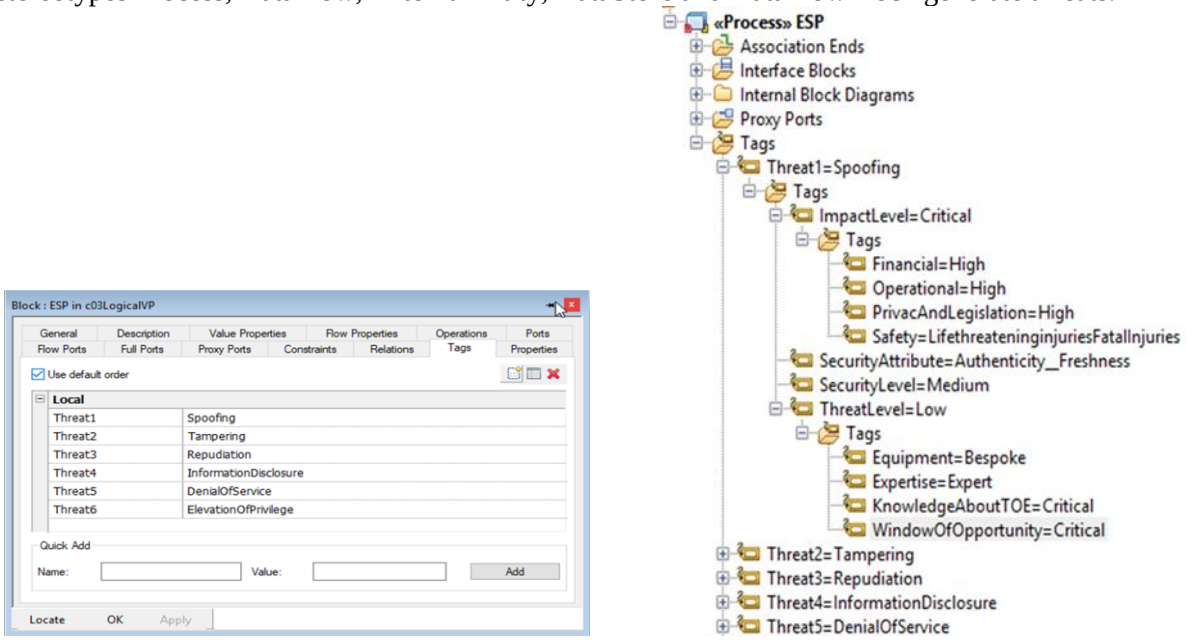

Figure 6: (a) Sample threats are listed and stored within the block ESP (left hand side) based on input of developer. (b) Impact level, threat level and security level for the threat are determined using helpers (right).

See Figure 6(b) for further illustration. The Threat1 Spoofing has the inner tag SecurityAttribute and the value is set to authenticity and freshness, see Table 1. The risk assessment step involves that the developer assigns the impact level and threat level to the threats based on the relevant parameters. The result of risk assessment is to derive a security level for the threat. The helper "Calculate Security Level" calculates impact level, threat level, security level and the values are assigned to corresponding tags, see Figure 6(b) based on the input of the developer.

Finally, the sample table view screenshot of Figure 7 gives an sample outlook on the cybersecurity requirements, its allocated assets, associated threats, security attributes and security levels, as determined by the developer.

Found 5 elements

ID	Specification	Asset	ThreatType	SecurityAttribute	SecurityLevel
CSR01	The connector shall provide integrity towards the stored data	To_MIL_Illumination	Tampering	Integrity	Critical
CSR02	The Confidentiality and privacy of the object flow to be preserved	To_MIL_Illumination	InformationDisclosure	Confidentiality__Privacy	High
CSR03	The authenticity of the ESP shall be ensured	ESP	Spoofing	Authenticity__Freshness	Medium
CSR04	The authorized users shall be able to use the ESP block to set the MIL_Heteronomy parameter whenever required	ESP	DenialOfService	Availability	Medium
CSR05	The Non repudiation and Freshness attributes to the Driver should be ensured	Driver	Repudiation	Non_repudiation__Freshness	Low

Figure 7: Table view of cybersecurity requirements.

An example of model checking application is given in Figure 8. The cybersecurity assessment relevant checks are as following. The cybersecurity requirement with empty Asset, SecurityLevel, SecurityAttribute, ThreatType is listed under Errors, since the assessment is not yet completed. Cybersecurity requirements with SecurityLevel value Critical are listed under Warnings. It is possible to implement any number of checks under safety and security using a similar procedure and programming approach as indicated. This ensures completeness and documentation as asked for in section 1 and section 2.

Roughly round 30 diagrams are used to perform the joint functional safety and cybersecurity assessment. Some model elements like blocks, tags, requirements, etc. which do not form a part of the listed diagrams, are also involved in the implementation: 3 BDDs, 3 IBDs, 1 UCDs, 12 ADs, 2 RDs, 2 Diagram views, 8 Table views, 32 Stereotypes and 4 Queries. The SysML is modeled in a single Rhapsody project which handles everything starting from system development, functional safety assessment and cybersecurity assessment. Note that the descriptive information that is normally provided (e.g. Safety/Security case) as document (typically in structured tables) can also be managed in the model using description fields, tags, attributes and comments, directly within diagrams. On the other hand, of course such summarizing tables can be generated from the prototype model extended with functional safety and IT-security SysML extensions as described. Thus, MBSE requirements as asked for in section 1 are met.

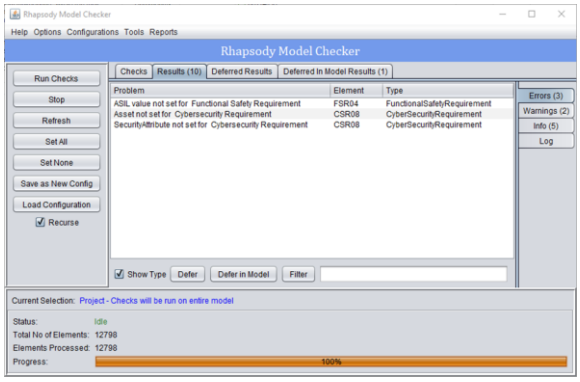

Figure 8: Model checker results regarding formal consistency and completeness of developer inputs for joint functional safety and cybersecurity assessments in Rhapsody model.

Inspection of effort estimates in Table 5 reveals that for real world applications, where MBSE SysML system models according to SPES can be assumed to be already existing, the extra effort for a SysML-supported joint functional safety and cybersecurity assessment by developers is rather limited.

Table 5: Effort of implementation and estimate of application in different phases in hours.

Implementation phase / Type of work	SysML modelling	Helper programming, Helper testing	Model checker development, Model checker testing
Implementation of approach including SysML profiles with stereotypes, tags, helpers, model checkers	35 h	126 h (programming)	63 h
Application to prototype	105 h	14 h (testing)	7 h
Real application of similar complexity	Only time needed for SysML model building according to approach	0 h	0 h

5. CONCLUSIONS

The advantages of the implementation for cybersecurity are as following: A seamless workflow is proposed and realized with extended SysML models, tags, stereotypes and helpers which support the

developer in cybersecurity assessment as well as automations and model checker based on the inputs. The completeness regarding cyber threats assessment by system developers is achieved in a formal manner in terms of fulfilling the HEAVENS process. The approach ensures traceability in all layers of system development using the SPES MBSE framework. The model checker and automations using helpers enhance the usability of the extended SysML models and increase their efficiency to document the HEAVENS assessment process conducted by system developers. The functional safety approach has been formalized and implemented in a similar way as the cybersecurity assessment process. The method implementation needs to be conducted only once and hence provides reusability. System development, functional safety and cybersecurity assessment go in parallel using a single model source.

The disadvantages of the approach include: There is the danger of formal conduction of functional safety and security assessment by just clicking at threats and entering assessments. Stereotypes supply many potential threats and the user needs to be careful to remove non-significant threats and to identify relevant threats, more generally more recent threat types might be lacking at all. The approach demands at least an intermediate level knowledge in SysML for functional safety and cybersecurity experts. Potentially not all useful SysML models needed for real system SysML modelling have been considered to describe systems completely.

The major implementation outcomes of the work included profiles for functional safety and cybersecurity which support the SPES profile and are conformal to it, with stereotypes, tags, model elements, helpers, model checker etc. The prototype SysML diagrams (the extended SysML model of the prototype), which used these outcomes, were also end products. They consisted of SPES based models in different layers and viewpoints. The visual paradigms were used to represent functional safety and cybersecurity relevant artefacts in the project. For instance, item, TOE, assets, functional safety concept, cybersecurity requirements, etc.

In summary, the methodology helps functional safety, cybersecurity and systems engineers to perform the human assessment in a better way compared to document-based approaches in terms of traceability, partial automation and management of complex project scenarios. Moreover, it was studied, how to effectively use SysML models to support international standards in automotive industry settings.

Some future scope and extensions feasible are listed. Straightforward extensions include: The item and threat categories within the HEAVENS approach could be (moderately) updated according to new IT architectures and a changing threat landscape, e.g. wireless sensor networks. In cybersecurity assessment within the HEAVENS approach, the technical cybersecurity concept creation and application deriving from the high level requirements as proposed within the approach could be implemented. Automatic document generation from SysML models is feasible (e.g. with the help of Rhapsody publishing engine interfaces [53]). One could allow for additional inputs and classifications of cyber threats, including counter measures. More complex extensions include: To identify options to make the approach a learning system in terms of additional potential inputs proposed to the developer and by challenging human input decisions, e.g. by asking for arguments that could be rated by other experts using the system, or to integrate a model-based attack tree (semi-automatic generation) for cybersecurity analysis.

Acknowledgements

Minor parts of the work have been supported by the German Federal Ministry for Economic Affairs and Climate Action (BMWE) founded project Real Driving Validation (RDV) Grant No. 19A21051D.

References

[1] J. A. Kim, S. Y. Choi, and S. M. Hwang, “Process & Evidence Enable to Automate ALM (Application Lifecycle Management),” in *2011 IEEE Ninth International Symposium on Parallel and Distributed Processing with Applications Workshops*, Busan, Korea (South), May. 2011 - May. 2011, pp. 348–351.

[2] S. Friedenthal, A. Moore, and R. Steiner, “Systems Engineering Overview,” in *A Practical Guide to SysML*: Elsevier, 2012, pp. 3–14. Accessed: 2012.
[3] S. Friedenthal, A. Moore, and R. Steiner, “Model-Based Systems Engineering,” in *A Practical Guide to SysML*: Elsevier, 2012, pp. 15–27. Accessed: 2012.
[4] M. Eigner, T. Dickopf, and H. Apostolov, “The Evolution of the V-Model: From VDI 2206 to a System Engineering Based Approach for Developing Cybertronic Systems,” in *Product lifecycle management and the industry of the future: 14th IFIP WG 5.1 International Conference, PLM 2017 : Seville, Spain, July 10-12, 2017 : revised selected papers*, 2017, pp. 382–393.
[5] *OMG Systems Modeling Language (OMG SysML™)*, OMG Systems Modeling Language, Nov. 2019. [Online]. Available: https://www.omg.org/spec/SysML/1.6/
[6] S. Friedenthal, A. Moore, and R. Steiner, “Getting Started with SysML,” in *A Practical Guide to SysML*: Elsevier, 2012, pp. 29–49.
[7] M. Larisch, A. Hänle, U. Siebold, and I. Häring, “SysML aided functional safety assessment,” in *Safety Reliablity and Risk Analysis: Theory, Methods and Applications, European Safety and Reliablity Conference (ESREL) 2008*, S. Martorell, C. G. Soares, and J. Barett, Eds., Valencia, Spain: Taylor and Franzis Group, London, 2008, pp. 1547–1554.
[8] M. Larisch, U. Siebold, and I. Häring, “Assessment of functional safety of fuzing systems,” in *International system safety conference // 27th International System Safety Conference and Joint Weapons System Safety Conference 2009 (ISSC/JWSSC 2009): Huntsville, Alabama, USA, 3 - 7 August 2009*, Huntsville, Alabama, USA: Curran, 2009.
[9] U. Siebold, M. Larisch, and I. Häring, “SysML modeling of safety critical multi-technological system,” in *European Safety and Reliablity Conference (ESREL) 2009*, R. Bris, C. G. Soares, and S. Martorell, Eds., Prague, Czech Republic.: Taylor and Franzis Group, London, 2009, pp. 1701–1706.
[10] U. Siebold, M. Larisch, and I. Häring, “Using SysML Diagrams for Safety Analysis with IEC 61508,” in *Sensoren und Messsysteme*, Nürnberg: VDE Verlag GmbH, 2010, pp. 737–741.
[11] A. Keis, “DDMS – Digital Design, Manufacturing and Service: Digitale Entwicklung am Beispiel Eurodrone: Presentation of Airbus Defense & Space, 29-th SafeTrans Industrial Day 2021,” Online, Dec. 13 2021.
[12] K. Pohl, H. Hönninger, R. Achatz, and M. Broy, Eds., *Model-Based Engineering of Embedded Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.
[13] K. Pohl, M. Broy, H. Daembkes, and H. Hönninger, Eds., *Advanced Model-Based Engineering of Embedded Systems: Extensions of the SPES 2020 Methodology*. Cham: Springer International Publishing; Imprint; Springer, 2016.
[14] *How a Tire Problem Became A Crisis for Firestone, Ford*: The Wall Street Journal. [Online]. Available: https://www.wsj.com/articles/SB965870212891028108
[15] *Major Recalls – Toyota Sudden Acceleration*: The Center for Auto safety. [Online]. Available: https://www.autosafety.org/major-recalls-toyota-sudden-acceleration/
[16] *Germany’s super train arrives - two years late*: www.thelocal.de. [Online]. Available: https://www.thelocal.de/20140218/germanys-new-super-train-ice-three-arrives-two-years-late-deutsche-bahn-siemens
[17] Chris Isidore, *Boeing has new safety problems with an older version of the 737 airplane*: CNN Business. [Online]. Available: https://edition.cnn.com/2019/10/10/business/boeing-737-ng-grounding/index.html
[18] *J3061 SURFACE VEHICLE RECOMMENDED PRACTICE Cybersecurity Guidebook for Cyber-Physical Vehicle Systems*, SAE International, Jan. 2016. [Online]. Available: http://www.sae.org/
[19] D. Mažeika and R. Butleris, “Integrating Security Requirements Engineering into MBSE: Profile and Guidelines,” *Security and Communication Networks*, vol. 2020, pp. 1–12, 2020, doi: 10.1155/2020/5137625.
[20] S. Kriaa, L. Pietre-Cambacedes, M. Bouissou, and Y. Halgand, “A survey of approaches combining safety and security for industrial control systems,” *Reliability Engineering & System Safety*, vol. 139, pp. 156–178, 2015, doi: 10.1016/j.ress.2015.02.008.
[21] Bosch Engineering Gmbh, *Bosch Engineering Gmbh website link.* [Online]. Available: https://www.bosch-engineering.com/de/
[22] *ISO 26262-1 Road vehicles - Functional safety - Part 1: Vocabulary,*, ISO, Switzerland, 2018. [Online]. Available: https://www.iso.org/standard/68383.html
[23] VDA QMC Working Group 13 / Automotive SIG, *Automotive SPICE Process Assessment / Reference Model*.
[24] *ISO/PAS 21448:2019: Road vehicles - Safety of the intended functionality*, 1, ISO, Jan. 2019. [Online]. Available: https://www.iso.org/standard/70939.html
[25] *Road vehicles — Cybersecurity engineering*, ISO/SAE 21434, Aug. 2021. [Online]. Available: https://www.iso.org/standard/70918.html

[26] D. Sales and L. Buss Becker, Eds., *2018 VIII Brazilian Symposium on Computing Systems Engineering (SBESC): Systematic Literature Review of System Engineering Design Methods*: IEEE, 2018.
[27] *ISO/IEC/IEEE 15288:2015 Systems and software engineering — System life cycle processes*, ISO. [Online]. Available: https://www.iso.org/standard/63711.html
[28] K. Höfig, M. Trapp, B. Zimmer, and P. Liggesmeyer, "Modeling Quality Aspects: Safety," in *Model-Based Engineering of Embedded Systems*, K. Pohl, H. Hönninger, R. Achatz, and M. Broy, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 107–118.
[29] M. Broy, W. Damm, S. Henkler, K. Pohl, A. Vogelsang, and T. Weyer, "Introduction to the SPES Modeling Framework," in *Model-Based Engineering of Embedded Systems*, K. Pohl, H. Hönninger, R. Achatz, and M. Broy, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 31–49.
[30] *TOGAF™ -- The Open Group Architecture Framework.* [Online]. Available: https://pubs.opengroup.org/architecture/togaf8-doc/arch/chap31.html
[31] Jeff A. Estefan, "Survey of Model-Based Systems Engineering (MBSE) Methodologies," vol. 25, no. 8, pp . 1–12, 2007.
[32] K. Pohl, H. Honninger, R. Achatz, and M. Broy, Eds., *Model-based engineering of embedded systems: The SPES 2020 methodology*. Berlin, New York: Springer, 2012.
[33] V. Sudheendran, "SysML supported functional safety and cybersecurity assessment for automotive MBSE: Master Thesis," IMTEK, University of Freiburg, Freiburg, 2020.
[34] Aljoscha Lautenbach and Mafijul Islam, *HEAVENS – HEAling Vulnerabilities to ENhance Software Security and Safety: Security Models*, 2nd ed. Sweden: The HEAVENS Consortium. [Online]. Available: https://www.vinnova.se/en/p/heavens-healing-vulnerabilities-to-enhance-software-security-and-safety/
[35] F. Swiderski and W. Snyder, *Threat modeling. Microsoft Press.*
[36] Deliverable D2.3, *Security requirements for automotive on-board networks based on dark-side scenarios*.
[37] CCRA Members, *Common Criteria for Information Technology Security Evaluation: Part 1: Introduction and general model.CCMB-2012-09-001*, 3rd ed.
[38] S. Christey, *PLOVER: Preliminary list of vulnerability examples for researchers: NIST Workshop Defining the State of the Art of Software Security Tools*.
[39] OMG, *About the Software & Systems Process Engineering Metamodel Specification Version 2.0: Software & Systems Process Engineering Metamodel.* [Online]. Available: https://www.omg.org/spec/SPEM/About-SPEM/
[40] Microsoft, *STRIDE.* [Online]. Available: https://docs.microsoft.com/en-us/azure/security/develop/threat-modeling-tool-threats
[41] Anton Bretting & Mei Ha, "Vehicle Control Unit Security using OpenSource AUTOSAR," Master's dissertation, Chalmers University of Technology ,University of Gothenburg, Gothenburg, Sweden, 2015. [Online]. Available: http://publications.lib.chalmers.se/records/fulltext/219822/219822.pdf
[42] IBM Rational Rhapsody, *Weblink.* [Online]. Available: https://www.ibm.com/support/knowledgecenter/SSB2MU_8.2.0/com.ibm.rhp.customization.doc/topics/rhp_c_ext_customized_programs_rhp.html
[43] Guillaume FINANCE, Objet Direct Analyst & Consultant, *SysMLModelling Language explained.* [Online]. Available: http://www.omgsysml.org/SysML_Modelling_Language_explained-finance.pdf
[44] S. Friedenthal, A. Moore, and R. Steiner, "Modeling Structure with Blocks," in *A Practical Guide to SysML*: Elsevier, 2012, pp. 119–183.
[45] S. Meacham, F. Gioulekas, and K. Phalp, "SysML based Design for Variability enabling the Reusability of Legacy Systems towards the support of Diverse Standard Compliant Implementations or Standard Updates: The Case of IEEE-802.15.6 Standard for e-Health Applications," in *Proceedings of the Eighth EAI International Conference on Simulation Tools and Techniques*, Athens, Greece, Aug. 2015 - Aug. 2015.
[46] S. Friedenthal, A. Moore, and R. Steiner, "SysML Language Architecture," in *A Practical Guide to SysML*: Elsevier, 2012, pp. 87–102.
[47] IBM Corporation, *Relationship types.* [Online]. Available: https://www.ibm.com/support/knowledgecenter/SS8PJ7_9.5.0/com.ibm.xtools.modeler.doc/topics/rreltyp.html
[48] S. Friedenthal, A. Moore, and R. Steiner, "An Automobile Example Using the SysML Basic Feature Set," in *A Practical Guide to SysML*: Elsevier, 2012, pp. 51–83.
[49] M. Daun, B. Tenbergen, and T. Weyer, "Requirements Viewpoint," in *Model-Based Engineering of Embedded Systems*, K. Pohl, H. Hönninger, R. Achatz, and M. Broy, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 51–68.
[50] A. Vogelsang, S. Eder, M. Feilkas, and D. Ratiu, "Functional Viewpoint," in *Model-Based Engineering of Embedded Systems*, K. Pohl, H. Hönninger, R. Achatz, and M. Broy, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 69–83.
[51] IBM Corporation, *Rational Rhapsody API.* [Online]. Available: https://www.ibm.com/support/knowledgecenter/SSB2MU_8.2.0/com.ibm.rhp.api.doc/topics/rhp_r_ext_using_rhapsody_api.html

[52] Bosch Mobility Solutions, *Electronic stability program.* [Online]. Available: https://www.bosch-mobility-solutions.com/en/products-and-services/passenger-cars-and-light-commercial-vehicles/driving-safety-systems/electronic-stability-program/
[53] IBM Corporation, *Integrating Rational Rhapsody and Rational Publishing Engine.* [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SS6RHZ_1.2.0/com.ibm.rational.pe.integration.doc/topics/c_rhapsody_integrate.html