Goal and Threat Modelling for Driving Automotive Cybersecurity Risk Analysis Conforming to ISO/SAE 21434

Christophe Ponsard, Valery Ramon and Jean-Christophe Deprez
CETIC Research Centre, Charleroi, Belgium

Keywords: Automotive, Safety, Security, Certification, Standards, Goal-oriented Analysis.

Abstract: As cars are increasingly connected and autonomous, they also become more exposed to cyber security threats. Providing strong protection and reactive response to such threats in a large industry involving many tiers and complex safety critical systems is challenging and required the development of the new ISO 21434 standard. Along with ISO 2626 dedicated to safety, it provides solid grounds for safety-security co-engineering. This paper focuses on how to provide effective and efficient support to the risk assessment phase based on a model-based approach. A rich goal-oriented meta-model is proposed to capture automotive assets and system properties, to estimate the impact of damage scenarios, to identify threats and to assess their feasibility. The approach is implemented as proof-of-concept through the meta-model adaptation of a generic co-engineering platform and is illustrated on the car light control sub-system.

1 INTRODUCTION

Cars have become totally connected objects and require communication to support a wide range of scenarios from information and entertainment purposes to new ways of operating the car itself such as driver-assistance, connected fleet and autonomous driving modes. This fast pace functional evolution has significantly increased the attack surface of the connected car (Shiho Kim, 2020).

Unfortunately, cyber security methods to protect the car assets did not improve at the same pace and relied, for a time, only on the J3061 set of best practices with little guidance on how to proceed (SAE, 2016). To fill this gap, ISO and SAE joined their effort to produce the ISO/SAE 21434 standard which will supersede J3061 by providing more guidance through actionable steps, a list of requirements for compliance, management processes and a global specification (ISO, 2020). This standard is still in draft form and due to be published in 2021.

The forthcoming ISO 21434 standard also complements the ISO 26262 by covering the cybersecurity perspective while ISO 26262 is addressing the functional safety perspective (ISO, 2011). The safety and security domains differ by the considered impacts: safety risks are related to consequences to people or environment, while cyber security risks are primary concerned with financial, operational, and privacy impacts. However, both share a common culture. From a risk management viewpoint, security risks are analysed through Threat Analysis and Risk Assessment (TARA) while safety performs Hazard Analysis and Risk Assessment (HARA). This opens the way to building a wider safety-security co-engineering approach, based on the fact safety has been part of the automotive development culture for decades. Given this evolution, it is anticipated that automotive suppliers will be asked to demonstrate compliance with ISO 21434 which can also lead to competitive advantage (Sembera, 2020).

Relying on model-based engineering is a realistic approach as most of the steps in the automotive development process are already model-based (Leopold, 2019). For example, SysML (OMG, 2005) and UML (OMG, 1997) are widely used as part of the architecture process or for model-based software development and integrated in automotive toolchains (Hause and Korff, 2007). Supporting model-based risk assessment is highly beneficial in order to support the systematic identification of assets, vulnerabilities, threats and countermeasures.

Our paper takes such a model-based approach for risk assessment. In order to consider both safety and security dimensions, we rely on a goal-oriented requirements engineering (GORE) framework able to structure systems properties and related risks (van Lamsweerde, 2009). Our contributions are to show

Ponsard, C., Ramon, V. and Deprez, J.
Goal and Threat Modelling for Driving Automotive Cybersecurity Risk Analysis Conforming to ISO/SAE 21434.
DOI: 10.5220/0010603008330838
In *Proceedings of the 18th International Conference on Security and Cryptography (SECRYPT 2021)*, pages 833-838
ISBN: 978-989-758-524-1

that such an approach supports:

- the identification of key assets, safety and security properties and related risks.
- co-engineering practices through commonly adopted methods to integrate TARA with HARA.
- a model-based tool chain with analysis, transformation and document generation.
- a validation on an automotive subsystem for controlling lights. capabilities.

This paper is structured as follows. Section 2 gives more background on the ISO 26262 and ISO 21434 standards. Section 3 presents our approach based on the modelling of assets, goals and hazards/threats and introduces our case study. Section 4 presents our HARA performed on a tool supported application of our method to an automotive case. Finally, Section 5 draws some conclusions and identifies further work.

2 BACKGROUND ON AUTOMOTIVE STANDARDS

This section gives minimal background about the content of the ISO 26262 standard and the new ISO/SAE 21434 standard.

2.1 ISO 26262 for Automotive Safety

The ISO 26262 "Road vehicles – Functional safety" is the international standard for functional safety of electrical and/or electronic systems in production automobiles (ISO, 2011). It is a risk-oriented standard, in the spirit of ISO 31000 (ISO, 2018). It is a specialisation of the IEC 61508, a risks basic functional safety standard which applies to all industries. The standard addresses the risk of hazardous operational situations using qualitatively HARA approaches resulting in the definition of safety measures to avoid or mitigate the effect of failures either of systemic or random nature.

From a process perspective, it follows the automotive W-shaped lifecycle combining two V cycles respectively for hardware and software design. HARA is performed during the concept phase and results in the identification of potential hazards. The corresponding risks are investigated with a rating function of the driving situation, the ability to control it and the severity of the caused harm. It is evaluated using an automotive safety integrity level (ASIL), which is assigned to each safety goal ranging from A to D for the most critical level. For systems with lower risks, quality management activities are deemed enough (Schmittner and Ma, 2015).

About cyber security, this standard defines the baseline cyber security guidelines for the development phase. It does not have specific requirements for post-production, decommissioning phases, automotive cyber security, or dealing with specific cyber security incidents.

2.2 ISO 21434 for Automotive Cyber Security

This upcoming standard, due to be released in 2021, aims at driving an industry-wide consensus related to key cyber security issues in the automotive domain (ISO, 2020). It will supersede the J3061 (SAE, 2016) good practices with more structured recommendations showing that the industry is fully embracing the challenges of ensuring automotive cyber security. Its scope is road vehicles (e.g. car, trucks, busses) and covers their sub-systems, components, connections, and software. Its purpose is to ensure that manufacturers and all participants in the supply chain have structured processes in place that support a "security by design" process.

For a process perspective, similarly to ISO 26262, it looks at the entire development process and life cycle of a vehicle. It follows the V-model and considers a wide range of activities such as TARA in the design branch, verification and validation in the testing branch and security monitoring, incident and response management in the operation phase.

The standard is structured in 10 sections and 15 clauses. It starts by defining (1) the scope, (2) the normative reference, (3) the glossary, (4) general considerations and (5/clauses 5-6-7) the management approach. Then (6/clause 8) focuses on risk assessment. It is followed by three sections respectively covering (7/clause 9) concept phase, (8/clauses 10-11) product development and (9/clauses 12-13-14) product, operation and maintenance. The final section (10/clause 15) deals with supporting processes.

Concerning the risk assessment, the standard does not impose any TARA methods but its clause 8 reminds about the mandatory steps it should cover, in the same spirit as the ISO 27005. It is depicted in Figure 1.

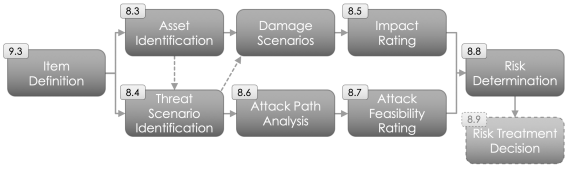

Figure 1: ISO 21434 TARA Process.

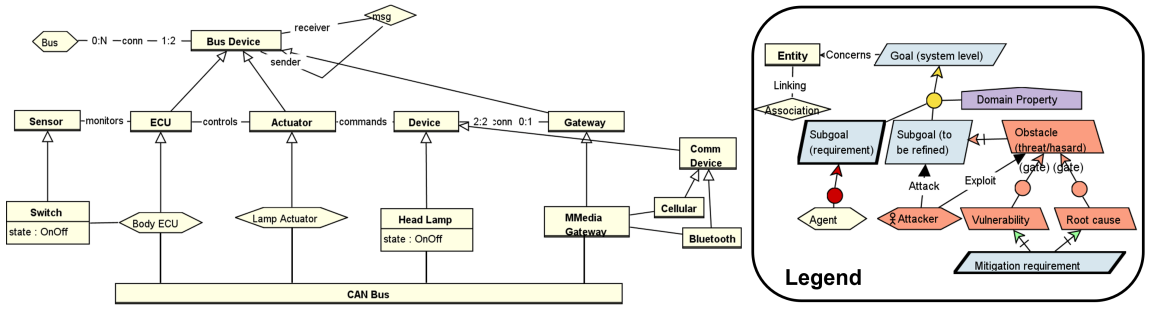

Figure 2: Object model of the light subsystem (with legend of notations).

3 GOAL-ORIENTED MODELS FOR CO-ENGINEERING

This section introduces our methodological framework based on GORE. Different variants of goal modelling were elaborated in the requirements engineering field such as i* (Yu and Mylopoulos, 1997), KAOS (van Lamsweerde, 2009), GSN (ACWG, 2018). This paper relies on KAOS. The required modelling notations are described in this section and depicted in the legend of Figure 2. More details are provided in the reference book (van Lamsweerde, 2009) and in a more focused application to co-engineering (Ponsard et al., 2021). To illustrate the approach, we use a light control subsystem. This case is actually used in the standard itself (ISO, 2020). Despite its limited size, it contains interesting issues to highlight the benefits of our approach. It is also analysed by method and tool developers (ASRG, 2020). Our modelling will be divided in three viewpoints: the object model capturing the domain, the goal model for properties and the obstacle model for risks.

3.1 Object/Asset Model

The object (or asset) model defines and interrelates all the concepts involved in a system. It must be able to capture all the vocabulary that is required to express properties (especially about safety and security) that will be structured in the goal model. It must also identify all valuable assets that could be threatened from a cyber security point of view.

This model captures domain concepts using entities, relationships, events or agents/attackers. The modelling notation used is close to the UML class diagram (OMG, 1997).

Figure 2 represents the asset model of our case study. The system is modelled using different abstractions that can be reused across many subsystems: Electronic Control Unit (ECU), sensors (the lamp switch in our case), actuators (lamp controller) and a variety of control devices (head lamps but also some communication devices). The communication is managed through an internal CAN bus on which messages are exchanged. A gateway may also ensure the bridge across subsystems, for example to connect communication devices to the CAN bus. Objects can have specific attributes reflecting important state information, e.g. the switch and lamp state.

3.2 The Goal Model

Goals capture, at different levels of abstraction, key properties the considered system should achieve (van Lamsweerde, 2009). The goal model structures goals in the form of AND-OR trees which express refinement relationships among goals. High-level goals can be progressively refined into more concrete and operational ones through relationships linking a parent goal to several sub-goals, with different fulfilment conditions using either “AND-refinement” (all subgoals need to be satisfied) or “OR-refinement” (a single sub-goal is enough, i.e. possible alternatives). Refinement can be characterised by some tactics which can help in checking their completeness and consistency, e.g. case-based (making design distinctions) or milestone-based (i.e. temporal steps). Goals can also bear specific attributes characterising specific nature, like safety of security which can be made graphically visible through specific decorators (SAFE/SEC).

Figure 3 shows the modelling of a safety property: lamp should be on at night (depicted by blue parallelograms). In our manual design, this relies on the fact the lamp is turned on if and only if the switch is activated and the assumption (depicted by a yellow parallelogram) the driver will activate and keep the switch on at night. Given the property (depicted by the little house) that switch can only be on or off (no auto-lightning considered), a case-based refinement is used to distinguish two possible state changes causes by turning the switching ON or OFF. The AND-refinement is represented by a single yellow cir-

cle connecting refined goals and useful properties that achieves an higher level goal. The “TurnOn WHEN SwitchON” goal is further refined using a milestone decomposition going through the whole communication chain: detecting the switch change by the ECU then sending a message on the CAN bus, processing it by the lamp controller to finally switch the lamp on. Different monitoring and controlling agents are involved in the process and are represented by yellow hexagons. Goals under the direct control of an agent are called requirements and have a thicker border.

3.3 The Obstacle Model

An obstacle represents an undesirable property. It is the dual concept of a goal and can be used to represent a safety hazard or a security threat. It is depicted by red parallelogram in the opposite direction than a goal. Obstacles can occur from the environment (i.e. a safety hazard) or be deliberately caused by an attacker (i.e. a security threat) like an attacker trying to take control of the light system. An attacker is depicted by a red agent. Like goals, obstacles can be refined using AND/OR trees, leading to decomposition structures quite similar to FTA in safety or AT in security. Refinement links can also be further characterised by concrete tactics through more specific safety/security gates (not illustrated in this paper). Leaf obstacles are either root causes (in safety) or vulnerabilities (in security). An obstacle is usually connected to the goal it is impacting through an obstruction relationship. Symmetrically an obstacle can be mitigated by an additional requirement. Attacker specific relationships also explicit the goal under attack or the vulnerability that is exploited.

Figure 4 illustrates obstacles on a partial attack decomposition unexpectedly turning the lamp on. It can be achieved by an attacker accessing the bus and

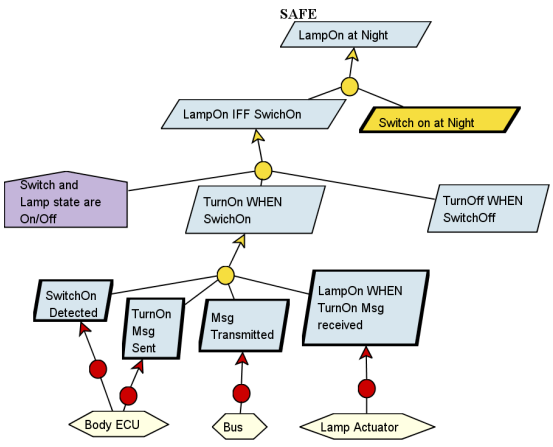

Figure 3: Goal model of the light sub-system.

spoofing a CAN message. A possible mitigation is to implement message integrity checks. Note the top level safety goal is not broken but a similar attack turning off the light at night would.

4 MODEL-BASED RISK ANALYSIS

In order to comply with the ISO 21434 TARA process, we will go through clauses 8.3 through 8.5 illustrated in Figure 1 using the goal modelling framework supporting safety and security co-engineering described in Section 3. The Objectiver platform (Respect-IT, 2005) is used as tool to build the model and perform the risk assessment but also as prototyping platform thanks to its plugin mechanism.

4.1 Asset Identification (8.3)

The asset identification step was already covered in Section 3.1 when presenting the object/asset model. Assets are identified by analysing the system goals. For example, the goal ”Lamp is On IFF Switch is On” enables to identify the Lamp and Switch. The goal refinement process conducts to the identification of the whole command transmission chain. Assets can also structured using more generic concepts forming the building blocks of a car (e.g. sensor, ECU, actuator) as shown in Figure 2. Those can then be instantiated in various subsystems.

Note the asset model can be further enriched with architectural information useful for the attack path analysis and later with components dealing with counter-measures (not detailed in this paper).

4.2 Damage Scenario

Damage scenarios can be elicited by trying to systematically break goals by introducing obstacles. It can be done on the different security dimensions of :

- *Confidentiality.* In this case, it is not relevant as everyone can see when lamps are turned on.
- *Integrity* results in obstacles related to unexpectedly turning lights on or off.
- *Availability* results in obstacles related to impossibility of turning lights on or off.

4.3 Threat Scenario Identification (8.4) and Attack Path Analysis (8.6)

This step is performed through attack trees which in

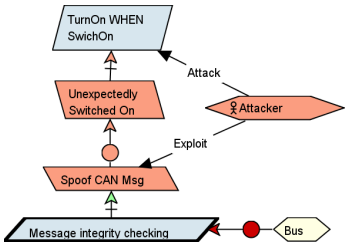

Figure 4: Obstacle (partial attack) on the light sub-system.

refinement of damage scenarios in a top-down approach or based on known vulnerabilities in a bottom-up approach. The later requires a richer architectural model which motivates to elaborate the object model as described in Figure 2. The full refinement of threats will naturally explore and discover different attack paths. In our case, the asset model analysis reveals a possible attack path using the media communication, either through the BT or cellular interface. From there, it can launch specific attack actions impacting either integrity or availability. The analysis can be performed on each of those security dimensions separately to be more modular. Figure 5 shows our analysis for integrity were both channels can be used to take control of the gateway and then some spoofed messages can be sent on the bus to make the lamp actuator believe the switch state has changed. This is depicted by a OR-refinement to compromise the navigation system, although the risk might be different for each path.

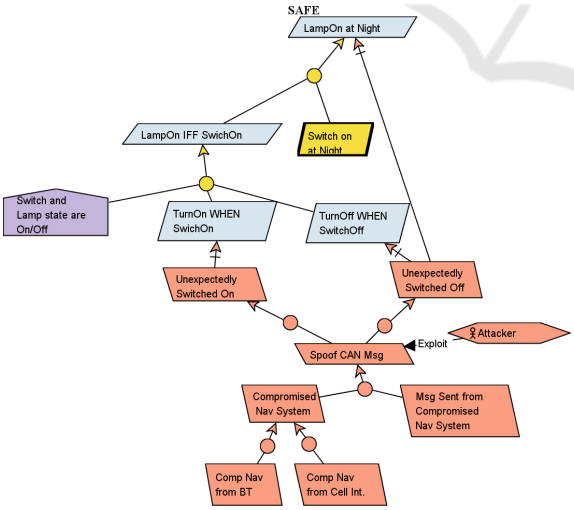

Figure 5: Analysis of the integrity attack paths.

4.4 Impact Rating (8.5)

The impacts have to be rated on four dimensions: Safety, Financial, Operational and Privacy (SFOP) (Sembera, 2020). In our case, there is no privacy impact given the lights are publicly observable. Operation impact is major as it changes the expected behaviour of the car. There is also no direct financial impact although the image damage might have an important indirect impact of this kind. About the safety impact, it can be rated in more details by reasoning on the safety impact of each threat w.r.t. the modelled safety goal stating lamps should be on at night:

- *R1 - Unexpectedly turning lamps on*: has no impact on the driver but may surprise another driver.
- *R2 - Unexpectedly turning lamps off*: can have a major impact if it happens at night.
- *R3 - Impossibility to turn lamps off*: has negligible impact although it can drain the battery.
- *R4 - Impossibility to turn lamps on*: has moderate impact because it requires to switch to a degraded mode, i.e. stop driving, at nightfall.

4.5 Attack Feasibility Rating (8.6)

To rate the feasibility of an attack, the process is first to evaluate each path. Attack factors can be used to rate each path against Time, Expertise, Knowledge, Opportunity and Equipment factors. To support this, we extended our tooling with the ability to capture those attributes using a meta-model extension. The various attack paths can be identified by looking at the different paths leading to leaf nodes in the attack tree. This can be implemented by a model query and results in the table show in Figure 6. The resulting table can be directly edited in our tooling.

The second step is to combine the factors depending on the way the paths are combined: AND-refinement will require to consider the minimal feasibility level while the OR-refinement needs to consider the maximal level. Note that other methods might be considered to deal with a richer set of refinement gates but are not elaborated here.

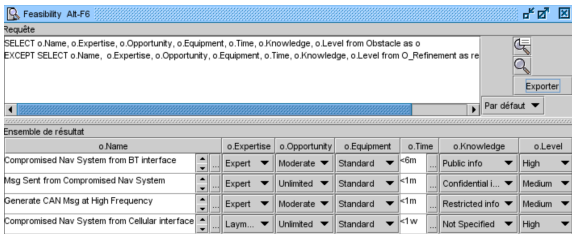

Figure 6: Feasibility analysis of the attack paths.

Based on this analysis, both the spoofing attack on integrity can be assessed to a medium level while the impact of the DOS attack on availability is low.

4.6 Risk Determination (8.9)

Based on the combination of the rating of impact and feasibility, it is possible to locate each risk in a qualitative risk matrix by using feasibility as column and impact as line, as shown in Table 1. This can be directly computed by the tool from information present in the model and then exported to a text processor table or spreadsheet (Ponsard et al., 2015).

Table 1: Safety Risk matrix for the light sub-system.

	Very Low	Low	Medium	High
Severe				
Major			R2	
Moderate		R4		
Negligible		R3	R1	

4.7 Risk Treatment Decision

Finally, actions must be taken to reduce the risks below an acceptable level. It can rely on tactics to accept, avoid, mitigate or transfer them. This step is not elaborated here due to space limitations.

5 CONCLUSION & NEXT STEPS

In this paper, we showed how to conduct a cyber security risk analysis in the automotive domain conforming to the new ISO 21343 standard. The proposed approach is model-based and also integrates with safety analysis in line with ISO 26262, opening the way to safety and security co-engineering. It was demonstrated on a generic goal-oriented toolset and illustrated on an automotive sub-system. Although limited in size, our case study could show the benefits of the approach related to the elicitation of threats, analysis of attack paths and assessment of risks. It also has good automation, scalability and reuse possibilities across sub-systems.

Based on this proof-of-concept, our next steps are to elaborate the risk treatment phase and to consider a larger case in the context of a on-going autonomous driving project. We also plan to improve our tool support and move to a domain specific system engineering tool, more adequate for integration and adoption in an automotive toolchain.

ACKNOWLEDGEMENTS

This work is partly funded by the CYRUS project of the Walloon Region (nr 8227).

REFERENCES

ACWG (2018). Goal Structuring Notation Community Standard, Version 2. The Assurance Case Working Group https://scsc.uk/r141B:1?t=1.

ASRG (2020). ISO21434 by Example. Automotive Security Research Group, https://www.youtube.com/watch?v=3LsNx-ljIK8.

Hause, M. and Korff, A. (2007). An overview of sysml for automotive systems engineers. *ATZelektronik worldwide*, 2.

ISO (2011). ISO 26262-1:2011 Road vehicles — Functional safety. https://www.iso.org/standard/43464.html.

ISO (2018). Iso 31000, risk management - guidelines, provides principles, framework. https://www.iso.org/iso-31000-risk-management.html.

ISO (2020). ISO/SAE FDIS 21434 Road vehicles — Cybersecurity engineering (draft). https://www.iso.org/standard/70918.html.

Leopold, D. (2019). Relevance of iso 21434 for the automotive development process. Itemis Blog https://blogs.itemis.com/en/relevance-of-iso-21434-for-the-automotive-development-process.

OMG (1997). Unified modeling language. http://www.omg.org/spec/UML.

OMG (2005). System modeling language. http://www.omg.org/spec/SysML.

Ponsard, C., Darimont, R., and Michot, A. (2015). Combining models, diagrams and tables for efficient requirements engineering : Lessons learned from the industry. In *Actes du XXXIIIème Congrès INFORSID, Biarritz, France, May 26-29, 2015*, pages 235–250.

Ponsard, C., Grandclaudon, J., and Massonet, P. (2021). A goal-driven approach for the joint deployment of safety and security standards for operators of essential services. *Journal of Software: Evolution and Process*.

Respect-IT (2005). The Objectiver Goal-Oriented Requirements Engineering Tool. http://www.objectiver.com.

SAE (2016). Cybersecurity Guidebook for Cyber-Physical Vehicle Systems - J3061_201601. https://www.sae.org/standards/content/j3061_201601.

Schmittner, C. and Ma, Z. (2015). Towards a framework for alignment between automotive safety and security standards. In *Computer Safety, Reliability, and Security*, pages 133–143. Springer.

Sembera, V. (2020). Iso/sae 21434 - setting the standard for connected cars' cybersecurity. Trend Micro Research.

Shiho Kim, R. S. (2020). *Automotive Cyber Security: Introduction, Challenges, and Standardization*. Springer.

van Lamsweerde, A. (2009). *Requirements Engineering - From System Goals to UML Models to Software Specifications*. Wiley.

Yu, E. and Mylopoulos, J. (1997). Enterprise modelling for business redesign: The i* framework. *SIGGROUP Bull.*, 18(1):59–63.