Towards an automated security-by-design approach in automotive system-of-systems architectures

Boris Brankovic
Josef Ressel Centre for Dependable System-of-Systems Engineering
Urstein Sued 1, A-5412 Puch, Austria
boris.brankovic@fh-salzburg.ac.at

Marco Ebster
Josef Ressel Centre for Dependable System-of-Systems Engineering
Urstein Sued 1, A-5412 Puch, Austria
mebster.its-m2020@fh-salzburg.ac.at

Katharina Polanec
Josef Ressel Centre for Dependable System-of-Systems Engineering
Urstein Sued 1, A-5412 Puch, Austria
katharina.polanec@fh-salzburg.ac.at

Christoph Binder
Josef Ressel Centre for Dependable System-of-Systems Engineering
Urstein Sued 1, A-5412 Puch, Austria
christoph.binder@fh-salzburg.ac.at

and Christian Neureiter
Josef Ressel Centre for Dependable System-of-Systems Engineering
Urstein Sued 1, A-5412 Puch, Austria
christian.neureiter@fh-salzburg.ac.at

***Abstract*—The development of future autonomous vehicles will become a challenging task, especially concerning the integration into a smart city context. Nowadays, vehicles must communicate with the surrounding environment to provide efficient driving features that prevent crashing and thus save passenger lives. However, the constant information exchange with the vicinity leads to growing attack surfaces of vehicles, which endangers the functional safety of vehicles in particular. Security-by-design seems to be a promising approach to overcoming security challenges and will become essential for the development of automotive architectures in the future. Therefore, the standard ISO-21434 was invented providing guidelines on how to tackle cybersecurity in the automotive context. One proposed method by this standard is the Threat Analysis and Risk Assessment (TARA) process used for analyzing cybersecurity threats. Nevertheless, no tools or approaches exist that provide full automation of the TARA process from an ISO-21434 perspective. Therefore, this paper proposes a concept to automate the TARA method by combining the security pattern engineering process with the Automotive Reference Architecture Model (ARAM) to enable a multi-layered security-by-design approach for the development of secure system-of-systems (SoS) architectures in conformity with ISO-21434.**

***Index Terms*—Automotive Cybersecurity, Smart City, System-of-Systems, ISO-21434, Model-Based Systems Engineering, V2X, Cyber-Physical Systems**

I. Introduction

According to [1] by 2025 every new car in advanced countries will be connected to a Global System for Communication (GSM), a connection typically providing access to the internet and backend systems. The demand for more connectivity in cars is especially driven by new features like Advanced Driving Assistant Systems (ADAS) and higher levels of automation. In general six autonomous levels are defined by the Society of Automotive Engineers (SAE) [2], where level zero specifies no autonomy and level five represents fully automated vehicles. However, to achieve full autonomy in cars more than a hundred electronic control units (ECUs) and a hundred million lines of code are required. For reference, the 2017 Ford F-150 has already exceeded 150 million lines of code and only partially covers ADAS functions - a number that will keep growing in the next years [3], [1], [4], [5]. Therefore, present and future vehicles with an autonomy level three or higher can be defined as connected autonomous vehicles (CAVs) and also as a specific kind of cyber-physical systems (CPS) due to their high diversity of interconnections and a great number of interrelated elements [6]. Moreover, the current communication landscape of CAVs is formed by vehicle-to-everything (V2X) communication, needed in different scenarios like *Vehicle Platooning*, or over-the-air (OTA) software updates [3], [7]. The increasing connectivity of CAVs implies also an increasing dependability as those systems cannot be seen as isolated systems anymore, but rather as system-of-systems (SoS), which leads to a diverse and immense attack surface as described in [8]. Recent research in [9], [10], [3], [11] has shown, that particularly in-vehicle networks using a Controller Area Network (CAN) bus as backbone are suffering from cybersecurity issues. Any compromise of these bus systems imperils the entire vehicle communication network and successful attacks could lead to serious malfunctions in the vehicle system e.g., failure of the braking system [3], [11].

Therefore, security must become an integral part of the entire system lifecycle of CAVs and vehicle security should follow a defense-in-depth strategy, using additional security techniques throughout the vehicle's architecture. Furthermore, research in [12], [13], [8] supports the importance of security-by-design, as it is significant to examine attack vectors and eliminate them prior to the development of systems. Consequently, with the introduction of the recently published ISO-21434 standard [14] for cybersecurity engineering in the automotive context, it is suggested to create a preliminary architecture for the identification of specific assets and their relationship in terms of cybersecurity. Hence, the authors in [15] made a first step towards the automation of the proposed Threat Analysis and

Risk Assessment (TARA) methodology in the ISO-21434. This novel approach makes use of so-called security patterns, which are collections of suitable security controls, that can be applied to mitigate threats. There are several approaches and tools to tackle this specific problem of integrating security already in the design process of system development. One of those is ThreatGet [16] which enables automated TARA. However, at the time of research, no projects are available that provide full automation of the TARA methodology from an ISO-21434 perspective. Thus, this work focuses on automation in the area of risk remediation, as well as verification of security patterns, to pave the way for an effective security-by-design approach.

To address all these aspects, this paper is organized as follows: In Section II, the related work about security engineering and security patterns, as well as relevant background on this topic is reflected. The implementation, application, and verification of the security patterns in their actual state are explained in Section III. Finally, in Section IV the results are outlined and a conclusion is given.

II. Related Work

A. Automotive Systems Engineering

The development of systems has become a challenging task, especially in the automotive domain as with the rising connectivity and the increasing interconnection to other systems, vehicles cannot be seen as isolated systems anymore, but rather as a SoS. Hence, interdisciplinary approaches such as systems engineering (SE) are needed to overcome the growing complexity within such systems and to enable the modeling and development of complete system architectures, utilizing proven concepts and thus satisfying the needs of stakeholders [17]. One specific SE framework that can be used to perform model based systems engineering (MBSE) tasks is the Automotive Reference Architecture Model (ARAM). This framework enables domain-specific SE in the automotive domain and provides with its three-dimensional structure the possibility to model complete CAV architectures from different perspectives and on various layers of abstraction. ARAM consists of five interoperability layers: *Business, Function, Information, Logical* and *Technical* - and allows the modeling of different aspects of a System of Interest (SOI), from the definition of requirements to the realization with suitable components [18].

B. System Security Engineering

In system security engineering (SSE), a security pattern documents the description of a solution to a frequent security problem in an individual context. In general, security patterns consist of five main attributes, which are *problem, context, forces, consequences, and solution*. The intent of the pattern i.e., solving recurring problems, is addressed by the first attribute *problem*. As every problem conveys a set of different challenges, security patterns must be tailored to the problems it is aimed to solve. The capability of identifying and analyzing security problems is a crucial feature of the security pattern engineering process [19]. The development of this kind of pattern starts by reviewing the system architecture and with the identification of security threats and vulnerabilities, using established security analysis methods such as threat modeling. Moreover, the application of the pattern results in a secure architecture as part of the security pattern engineering process [20].

C. ISO 21434 - Cybersecurity Engineering

The increasing complexity of road vehicles requires an appropriate standard with definitions for establishing, maintaining, and analyzing automotive cybersecurity. This requirement is encountered by the international standard ISO-21434, which specifies process requirements and guidelines for securing automotive architectures [14]. The standard addresses cybersecurity in the automotive context so that the development of electrics/ electronics (E/E) systems can keep up with the state-of-the-art and evolving attack methods. Moreover, the standard describes fifteen parts, divided into the description of initial aspects of cybersecurity and technical components. The latter is the most significant for this work, as it yields information about *continuous cybersecurity activities* and *TARA methods*. Those deal with actions and procedures that can be utilized during the life cycle of a product to guarantee item security. TARA includes methods for assessing the resulting risks to an item and analyzing threats. This very method is described in a seven-step process, starting from the identification of assets and threats to the impact rating and risk treatment decision. Following this process results in the specification of cybersecurity goals to an overall cybersecurity concept, which defines the requirements to be fulfilled [14], [8].

III. Implementation

A. Security Pattern Development

A first example of the development of security patterns is explained in [15], where the authors use a suitable case study about an OTA software update scenario as basis for the security design process. This case study is modeled with the ARAM framework and implemented as domain specific language (DSL) for the modeling software Enterprise Architect (EA). Figure 1 shows an example pattern for the core component *CAV Central Gateway*, which was created with the ARAM Toolbox (https://aram.dsse.at/), an add-in for EA.

In general, the security pattern is based on the current state of the ARAM Information Layer. With this approach, it is possible to integrate security design principles in an early stage of architecture development and it enables a top-down refinement of the security pattern on each layer. The pattern can be constructed with a four-step process:

- Identification of the problem space and scope.
- Threat modeling.
- Definition and mapping of security control objectives.
- Assembling the security pattern.

Following these guidelines results in a security pattern that is in accordance with the ISO-21434 standard, as this four-step principle leans on the TARA process. Tools like ThreatGet

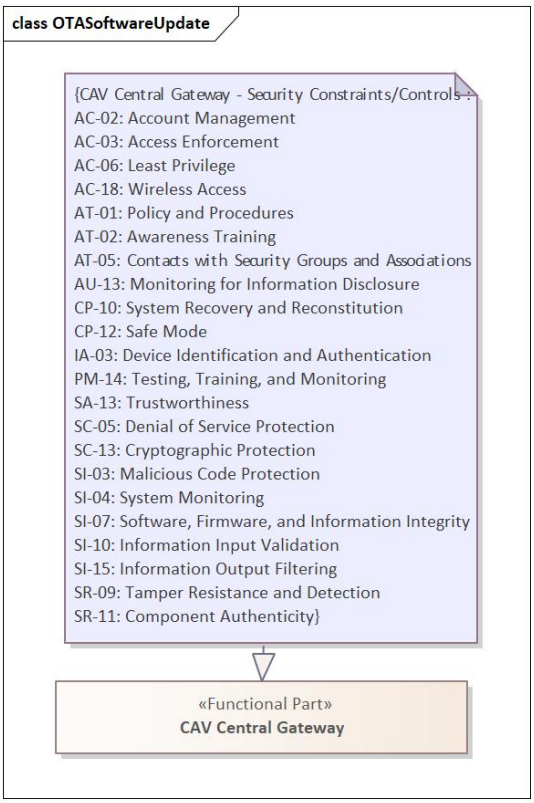

Fig. 1: Example security pattern for an OTA core component.

[16] are used within this approach to accomplish the threat modeling step and to analyze potential threats and risks in the modeled architecture. However, there is still a lack in automating the entire TARA method as no tools exist that provide (i) a list with suitable security controls used to mitigate the occurring threats in an automotive cybersecurity context and (ii) an automation of those security controls i.e., applying the controls with an additional verification process.

B. ISO 21434 extension for the ARAM Toolbox

Automated security-by-design concepts are needed for the purpose of the creation of user-friendly and easy-to-use security design processes, enabling complete automation of the TARA process. To achieve this, the ARAM Toolbox, an add-in for the modeling software EA and the ARAM DSL were extended to support the application of security patterns, the automation of associated security controls and the verification against ISO-21434. ARAM makes use of the EA add-in model for customization and extension of the EA user interface. The most significant feature *Apply Security Controls* is explained in more detail.

Apply Security Controls **-** This functionality is used to automate security patterns, that have already been applied on previous ARAM layers such as the ARAM Information Layer. For example, a security pattern must be applied on the Information Layer of ARAM, so that the automation affects the lower Technical Layer of ARAM. If these prerequisites

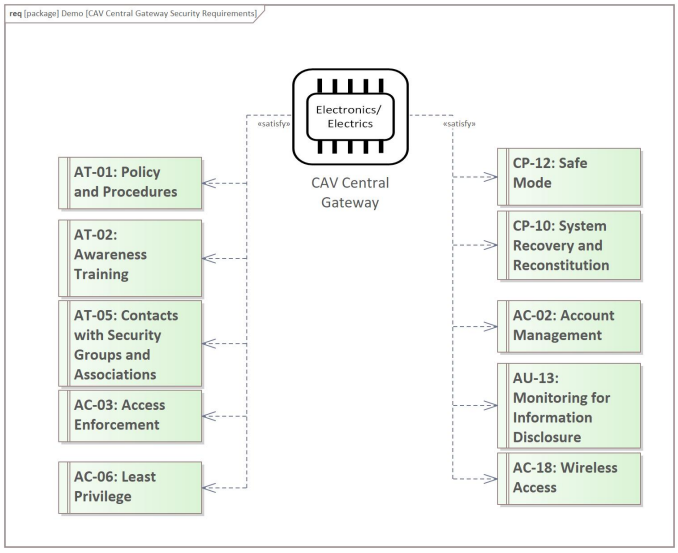

Fig. 2: SysML requirements diagram for the verification of security patterns.

are fulfilled, the logic will start searching for existing security patterns (e.g., Figure 1) in the ARAM structure and reads the related security controls, defined for each core component (hereinafter referred to as assets). The security controls are then accordingly mapped to security components and linked to the modeled assets on the ARAM Technical Layer. As the final step, for each asset, a Systems Modeling Language (SysML) requirements diagram is created (illustrated in Figure 2), which represents high-level requirements that each security component satisfies for its asset and where each requirement in turn associates to a security control originating from the applied pattern. This procedure is a crucial aspect with respect to the verification according to ISO-21434.

C. Automation & Verification

Furthermore, a verification of the solution is required to accomplish an ISO-21434-compliant cybersecurity concept. The model of the OTA architecture is used as starting point for the evaluation. As previously stated, threat modeling identifies threats for each asset which in turn is addressed by appropriate security controls as part of the security pattern engineering process. This results in a security control list based on the National Institute of Standards and Technology (NIST) SP 800-53 Rev. 5 standard [21], which each considered asset must implement to mitigate the identified threats. As the assets themselves are not able to implement these security controls, additional components, also referred to as security components, are used for the implementation of the required security controls for its assets. In the course of this procedure, a SysML requirements diagram, shown in Figure 2, is additionally added for each asset as a child diagram, containing high-level requirements satisfied by the respective asset. If the security controls have addressed all threats, the considered OTA architecture can be considered secure against

the identified threats. Thus, if all security controls defined in the security pattern have been implemented for each asset, the overall cybersecurity goal can be successfully verified against ISO-21434. This procedure can be performed with the ARAM Toolbox feature *Verify Implementation of Security Controls*.

IV. Discussion, Conclusion & Future Work

The level of connectivity and the amount of hardware, as well as software components within CAVs are increasing significantly. This comprises interactions with remote systems (e.g.,V2X) and a rising number of internal networks. Hence, CAVs are evolving always more to SoS, which boosts the dependability and with that the attack surface of this kind of vehicle. Thus, the international standard ISO-21434 was developed to tackle these concerns raised by growing system complexity and the subsequent cybersecurity challenges. However, ISO-21434 provides a standard for automotive cybersecurity engineering, but it does not provide automated methods in terms of saving more effort and time in creating secure system architectures. While the work in [15] proposes a first concept of integrating the TARA process into the ARAM framework through security patterns, this paper presents a method of automating these processes, to enable simultaneous engineering of automotive cybersecurity features. Accordingly, the current state of the TARA automation as an extension of the ARAM Toolbox add-in is presented, which automatically applies security controls for each asset and integrates these into the system architecture. This results in an additional number of security components for each considered asset. Further, these components can be traced within the TARA process, which allows a transparent assessment of the security relevance of the created architecture. The clear link between threat identification and mitigating controls allows an easy-to-use and traceable verification concept of whether the security goal is met in automotive SoS architectures. Furthermore, as the ISO-21434 standard is limited to the system boundaries of vehicles, another standard such as ISO-24089 [22] must be considered as well. This standard provides requirements and guidelines related to software update engineering.

In future projects, the existing security pattern will be adapted to the ISO-24089 standard and enhanced to support also security-by-design in the context of smart cities. With these intentions, an effective solution for the development of secure SoS architectures can be achieved and used to analyze cybersecurity weaknesses and impacts in a V2X environment.

Acknowledgment

The support for valuable contributions of Robert Bosch GmbH is gratefully acknowledged. The financial support by the Austrian Federal Ministry for Digital and Economic Affairs and the National Foundation for Research, Technology, and Development, and the Christian Doppler Research Association as well as the Federal State of Salzburg is also gratefully acknowledged.

References

[1] D. P. Möller and R. E. Haas, *Guide to Automotive Connectivity and Cybersecurity: Trends, Technologies, Innovations and Applications*, 1st ed. Springer Publishing Company, Incorporated, 2019.

[2] O.-R. A. D. O. Committee, *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*, SAE International, apr 2021.

[3] L. L. Bello, R. Mariani, S. Mubeen, and S. Saponara, "Recent advances and trends in on-board embedded and networked automotive systems," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 1038–1051, 2019.

[4] K. Pohl, H. Hönninger, R. Achatz, and M. Broy, Eds., *Model-Based Engineering of Embedded Systems: The SPES 2020 Methodology*. Heidelberg: Springer, 2012.

[5] P. Karkare, "Model-based systems engineering for autonomous vehicle development," Siemens PLM Software, Tech. Rep., 2018.

[6] A. Chattopadhyay and K.-Y. Lam, "Security of autonomous vehicle as a cyber-physical system," in *2017 7th International Symposium on Embedded Computing and System Design (ISED)*, 2017, pp. 1–6.

[7] W. Böhm, M. Broy, C. Klein, K. Pohl, B. Rumpe, and S. Schröck, Eds., *Model-Based Engineering of Collaborative Embedded Systems : Extensions of the SPES Methodology*. Cham, Switzerland: Springer, 2021. [Online]. Available: https://publications.rwth-aachen.de/record/810259

[8] D. Ward and P. Wooderson, *Automotive Cybersecurity: An Introduction to ISO/SAE 21434*, 2021, pp. i–xii.

[9] P. Kleberger, T. Olovsson, and E. Jonsson, "Security aspects of the in-vehicle network in the connected car," in *2011 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2011, pp. 528–533.

[10] J. Liu, S. Zhang, W. Sun, and Y. Shi, "In-vehicle network attacks and countermeasures: Challenges and future directions," *IEEE Network*, vol. 31, no. 5, pp. 50–58, 2017.

[11] Y. Xie, Y. Zhou, J. Xu, J. Zhou, X. Chen, and F. Xiao, "Cybersecurity protection on in-vehicle networks for distributed automotive cyber-physical systems: state-of-the-art and future challenges," *Software: Practice and Experience*, vol. 51, no. 11, pp. 2108–2127, 2021.

[12] L. Reger, "The road ahead for securely-connected cars," in *2016 IEEE International Solid-State Circuits Conference (ISSCC)*. IEEE, 2016, pp. 29–33.

[13] K. Kim, J. S. Kim, S. Jeong, J.-H. Park, and H. K. Kim, "Cybersecurity for autonomous vehicles: Review of attacks and defense," *Computers & Security*, vol. 103, p. 102150, 2021.

[14] International Organization for Standardization, *ISO 21434:2021 Road Vehicles - Cybersecurity engineering*, Std., 2021.

[15] M. Ebster, B. Brankovic, K. Polanec, C. Binder, and C. Neureiter, "Towards a security-by-design approach enabling automated validation in automotive architectures," in *13th Complex Systems Design & Management conference*, Paris, France, 2022.

[16] C. Schmittner, S. Chlup, A. Fellner, G. Macher, and E. Brenner, "Threat-get: Threat modeling based approach for automated and connected vehicle systems," in *AmE 2020 - Automotive meets Electronics; 11th GMM-Symposium*, 2020, pp. 1–3.

[17] R. Guillerm, H. Demmou, and N. Sadou, "Engineering dependability requirements for complex systems - A new information model definition," in *2010 IEEE International Systems Conference*, April 2010, pp. 149–152.

[18] K. Polanec, J.-A. Gross, B. Brankovic, and C. Neureiter, "Evolution of the automotive reference architecture model towards a domain-specific systems engineering approach," in *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2022, pp. 1–4.

[19] H. Mouratidis, "Software engineering for secure systems: Industrial and research perspectives: Industrial and research perspectives," 2010.

[20] H. Martin, Z. Ma, C. Schmittner, B. Winkler, M. Krammer, D. Schneider, T. Amorim, G. Macher, and C. Kreiner, "Combined automotive safety and security pattern engineering approach," *Reliability Engineering & System Safety*, vol. 198, p. 106773, 2020.

[21] J. T. Force, "Security and privacy controls for information systems and organizations," National Institute of Standards and Technology, Tech. Rep., 2020.

[22] International Organization for Standardization, *ISO 24089:2023 Road Vehicles - Software update engineering*, Std., 2023.