EMILIJA KASTRATOVIĆ

COMPARING DIFFERENT VEHICLE ARCHITECTURES BASED ON ATTACK PATH ANALYSIS

COMPUTER NETWORKS AND IT SECURITY
PROJECT PROPOSAL

Supervisor: Michael Wolf

Institut für Verteilte Systeme
Institute of Distributed Systems

University of Ulm
uulm.de/in/vs
February 2, 2023

ABSTRACT

The increased use of technology in modern vehicles has made cybersecurity a crucial part of the development process of modern vehicles, making it more and more critical for the safety and security of the vehicle and the privacy and personal data of drivers and passengers.

The internal vehicular network of such E/E systems plays a vital role in the overall cybersecurity of a modern vehicle. For example, attackers might exploit potential attack paths in the network to gain unauthorized access to a vehicle's systems or networks.

However, most security testing is done at later stages of development, when it is more complex and costly to address vulnerabilities. Additionally, due to the nature of security testing, it is carried out manually by experts with a high level of expertise and experience with other cybersecurity tools and techniques. These factors result in a stagnant security testing process that cannot maintain pace with the increasing complexity of modern vehicles.

This thesis focuses on comparing various vehicular network architectures in terms of which offers more security based on their attack path feasibility. A comprehensive evaluation of multiple architectures is conducted by introducing a tool that integrates the generation of attack paths with an evaluation of each network architecture. The results offer a guiding beacon for future development and evaluation of vehicular network designs.

Emilija Kastratović: *Comparing Different Vehicle Architectures Based on Attack Path Analysis,* Computer Networks and IT Security
Project Proposal, © February 2, 2023.

WEBSITE:
uulm.de/in/vs

E-MAIL:
emilija.kastratovic@uni-ulm.de

CONTENTS

1 INTRODUCTION

1.1 RESEARCH FIELD: CYBERSECURITY OF VEHICULAR NETWORKS

Modern vehicles are becoming increasingly reliant on technology, with a wide range of systems and components being connected to the internet and each other. This includes everything from infotainment systems and navigation to advanced driver assistance systems and autonomous driving features. ISO 26262 describes these so-called "Electronic/Electrical (E/E) Systems" as systems that consist of electrical and electronic elements and components. Examples include Electronic Control Unit (ECU)s, sensors, actuators, connections, and communication systems like Controller Area Network (CAN), Ethernet, and Bluetooth [1]. As these technologies become more and more important, strong cybersecurity measures are also becoming increasingly important. Cybercriminals are constantly finding new ways to exploit vulnerabilities in these systems which can have serious consequences for users, which includes their safety, privacy, finances, and operational viability [2]. Ensuring the safety and security of connected vehicles is crucial to the success of the emerging world of connected and autonomous transportation.

The internal vehicular network architecture plays a crucial role in the overall cybersecurity of a modern vehicle as it determines how different vehicle systems and components are connected and communicate. Attack Paths play an important role in vehicle networks and security because they define the specific routes that a malicious actor might use to attack a vehicle's systems or networks. A well-designed internal vehicular network architecture can help minimize cyberattack risk. With the increasing number of systems and components that are connected to the internet and each other, the complexity of the internal vehicular network architecture is also increased. This can make it more challenging to design and implement effective cybersecurity measures as more potential points of vulnerability arise that need to be addressed. Therefore, companies developing connected and autonomous vehicles need to prioritize cybersecurity in designing their internal vehicular network architecture, which can help to ensure the safety and security of the vehicle, as well as protect the privacy and personal data of drivers and passengers.

Security testing is, therefore, a crucial part of the development process. A Threat Analysis and Risk Assessment (TARA), for example, is performed early during the development process to identify and prioritize potential risks. This information can be used to guide the design and implementation of controls or countermeasures to reduce or mitigate these risks. Once the system is developed, e.g. a penetration test is carried out to validate the effectiveness of these controls and to identify any remaining vulnerabilities. The results of the pentest can then be incorporated back into the TARA process to update the risk assessment and prioritize future risk mitigation efforts. For example, companies can implement appropriate security measures by understanding the attack paths that might be used to gain unauthorized access to a vehicle's systems. These include encryption, authentication protocols, and firewall systems, etc. to protect against cyber threats.
In this way, the combination of a pentest and TARA provides a complete and itera-

tive approach to security assessment.
However, security testing is often carried out in the late stages of development, which can lead to the discovery of vulnerabilities at a time when it is more difficult and costly to address. Additionally, they are considered to be a skill-based activity that is still carried out manually. It requires a high level of expertise and experience with other cybersecurity tools and techniques. The increased complexity of modern vehicles and the arduous nature of state-of-the-art security testing methods make it more unfeasible for companies to conduct them as is done now.
Thus, the need for a more efficient approach to improve overall security testing is evident. By automizing the process of evaluating automotive network architectures and their attack paths, potential vulnerabilities can be assessed early on in the development process resulting in overall more efficient security testing, improving the flexibility, costs and accelerateing it.

1.2 BACKGROUND AND MOTIVATION

As a computer science student with a passion for cybersecurity and a focus on cybersecurity in my studies, I joined the CarIT Security team at Mercedes-Benz Tech Innovation (MBTI) a year ago as a working student. I worked with automotive protocols such as CAN, CAN FD, FlexRay, and Ethernet, used software like CANoe, DTS, and proprietary tools for pentesting, performed scans, and created architecture diagrams of vehicular networks. Through my work there, I got to know the field of vehicular cybersecurity.

MBTI is facing the same challenges mentioned in 1.1 and is searching for a solution to this problem. As already described, the internal vehicular network architecture plays a crucial role in the overall cybersecurity of a modern vehicle. My supervisor and colleague proposed the topic of automated vehicular network evaluation to address the company's need for a tool that can aid in the security testing process. By automatically evaluating vehicular network architectures, the same benefits as described in 1.1 can be achieved.

2 STATE OF THE ART

There are various approaches to assessing the security of vehicular networks. Cybersecurity standards and frameworks give guidance and best practices for designing, implementing, and testing the cybersecurity of automotive systems and networks. The following standards are mentioned in virtually every piece of literature; thus, they are the most important ones to consider. Moreover, they offer a basis for the thesis itself as well as the development of a tool to assess the security of vehicular networks, as their further use is intended to be used in conjunction with these standards and frameworks to ensure compliance with them:

ISO 26262 is an international standard for ensuring the functional safety of E/E systems in vehicles. It provides a framework for the development and management of safety-related systems throughout the lifecycle of a vehicle, from design to operation. The standard is divided into several parts, each addressing a specific aspect of functional safety. It aims to help organizations manage the functional safety of their systems, ensure they meet a defined level of safety, and demonstrate that the systems are safe for the intended use. ISO 26262 helps organizations identify and mitigate potential hazards, reducing risks to an acceptable level. It provides a systematic approach to functional safety, ensuring the safety of vehicles and their passengers.

ISO 21434 is an international standard for the execution of functional testing and specifies engineering requirements for cybersecurity risk management regarding the lifecycle of electrical and electronic E/E architecture systems in road vehicles, including their components and interfaces [2]. ISO 21434 outlines the security requirements, goals, and measures to consider throughout the entire lifecycle of a vehicle, including design, development, production, and operation. It aims to assist organizations in managing the security of their vehicles and ensuring they meet a defined level of security by providing a systematic approach to identifying and evaluating security-related risks and implementing measures to reduce them to an acceptable level. The standard is intended to be used in conjunction with ISO 26262, which focuses on functional safety, to provide a comprehensive approach to ensuring the safety and security of vehicles. It also explains how to integrate security into the functional safety process, helping organizations manage security risks similarly to how they manage functional safety risks.

SAE J3061 is a guide for cybersecurity best practices for the automotive industry and was created based on existing methods. The guide provides a comprehensive set of best practices for securing automotive systems and vehicles from cyber threats and covers various aspects of cybersecurity, including threat modeling, risk management, security testing, incident response, and security management. They are intended to be flexible, pragmatic, and adaptable in their further application to the vehicle industry and other cyber-physical vehicle systems. It provides a framework for organizations to incorporate cybersecurity into the lifecycle of vehicle systems, information on standard tools and methods used in designing and verifying systems, basic principles for cybersecurity, and a foundation for further standards development. SAE J3061 is intended to be used in conjunction with other standards

and guidelines, such as those mentioned above [3].

AUTomotive Open System ARchitecture (AUTOSAR) is a standard for the development of software for ECUs in the automotive industry [4]. The goal of AUTOSAR is to create and establish an open and standardized software architecture for automotive ECUs that is scalable to different vehicle and platform variants, transferable of software, considering availability and safety requirements, collaboration between various partners, sustainable use of natural resources, and maintainable during the product lifecycle. This improves the efficiency of development, enables the integration, exchange, reuse, and transfer of functions within a vehicle network, and helps manage the growing complexity of technology and economics in automotive software development.

The following frameworks also, are mentioned frequently in literature and offer a base for the development of a tool to assess the security of vehicular networks:

TARA is an essential process for ensuring the cybersecurity of systems and networks, especially in the automotive industry. In the context of automotive systems, a TARA can be used to identify and evaluate potential threats and vulnerabilities to the electronic and electrical (E/E) architecture of vehicles. This includes identifying assets such as connected systems and networks and evaluating the likelihood and potential impact of threats such as cyberattacks, hacking, and software vulnerabilities as well as determine appropriate countermeasures to mitigate these risks. In addition, it can help prioritize these threats and vulnerabilities based on their potential impact on safety, financial, operational, and privacy aspects.
An attack path analysis can be an important component of TARA, helping to identify and evaluate the potential pathways that an attacker might use to gain unauthorized access to the vehicle's system. Based on such analysis, appropriate changes can be made to the vehicle's network architecture as a countermeasure. The TARA process can be used to ensure compliance with industry standards, such as ISO 26262 or ISO 21434, and can be integrated with other frameworks, like HEAVENS or EVITA. Regularly reviewing and updating the TARA process is crucial to keep up with the evolving threats and vulnerabilities in the automotive industry, which is constantly evolving with the integration of new technologies such as connected cars, autonomous driving, and V2X communication [5].

Important frameworks include HEAling Vulnerabilities to Enhance Software, Security, and Safety (HEAVENS), and E-safety vehicle intrusion protected applications (EVITA). HEAVENS performs risk assessments of general IT systems and models explicitly built for automotive systems and uses threat and impact levels to calculate risks [6]. The primary objective of the framework is to identify security requirements and vulnerabilities in automotive systems and to provide countermeasures to minimize the risks associated with these vulnerabilities. It uses the Microsoft STRIDE model for threat modeling and aligns its impact level estimation parameters with established industry standards such as ISO 26262. It is a great candidate as a framework for automotive risk assessments over traditional IT risk assessment models.
The EVITA framework, which essentially performs the same things as HEAVENS, but also considers the potential of attacks to impact the privacy of vehicle passengers, financial losses, and the operational capabilities of the vehicles systems and functions [7].

Many of these approaches aim to standardize the process of assessing the security of vehicular networks. However, most of them are based on manual penetration testing and manual vulnerability assessment today. This is because penetration testing is an experienced-based and explorative skill that is difficult to automate. Further research aims to improve or couple existing approaches like performing a TARA, as well as automate and accelerate the process of security testing.

F. Sommer et al. introduce the concept of Model-Based Security Testing using an Extended Finite State Machine (EFSM) model [8]. The Automotive Security Model section describes an E/E Architecture, security measures, and further development artifacts to protect vehicles against attacks. The model is based on the EFSM, with nodes representing attacker privileges and transitions representing a vulnerability. The Proof of Concept section of the paper demonstrates how the model can be used to identify different Attack Paths, analyze the model for attack paths, and execute the attack paths on a real vehicle. The incremental approach allows for the redefinition of attack paths, making it useful at different stages during development. The paper concludes that this approach can be helpful in identifying attack paths and potential vulnerabilities in automotive systems, thus helping to improve the security of vehicles.

F. Sommer et al. further expand on the same model-based approach by using a database of successful vehicular penetration tests [9]. The attack path generation process involves using an attack database, which describes vulnerabilities and exploits that can be used, including attack taxonomy and classification, attack steps, requirements, restrictions, components, and interfaces. The database can also be used to find new attack paths by permuting existing attack steps. Additionally, the process of creating the database can be done iteratively over several penetration tests and can be transferred to test scripts. However, there is a risk that the attack path generated may not be transferable, which can be circumvented by permuting previous attacks. The authors also suggest that further testing activities, such as black-box testing, should be carried out. Despite its limitations, this approach can be used as a useful tool to automate the process of penetration testing and improve the efficiency of security testing in the automotive industry.

J. Dürrwang et al. further describe the concept of attacker privileges mentioned in the papers above [10]. Several types of privileges that an attacker may seek to gain access to a vehicle's communication system and components are defined, such as "Read/Write," "Execute," "Read," "Write," and "Full Control.". They note that channels that are not protected by security measures can be immediately accessed by an attacker, but interpretation is needed in other cases. It also mentioned that the attacker needs to reach one of these privileges to access further attached communication systems and components. The authors applied their privilege model to real-world automotive security attacks to demonstrate its practical use, in which an attacker is connecting to the vehicle via the On-board diagnostics (OBD) port, which is connected to the central gateway via CAN, and then uses the central gateway to gain access to the internal vehicle network. They also showed the automatic generation of attack trees using a model checker in a custom software tool and an application of their privileges in security testing by describing attack paths. In future work, the authors plan to formalize the security testing approach to allow for early testing during development and to evaluate the TARA and security testing approach in a case study.

In contrast, D. Zelle et al. introduce a concrete approach, "ThreatSurf" [11], which presents an algorithm for automated generation and rating of attack paths in the automotive industry, using various attack building blocks and assessing attack feasibility. The attack feasibility assessment can be used in a TARA to assess an entire attack path of a threat scenario. It also discusses different methods for calculating attack paths, such as Sum, Average, Maximum, and Hybrid-weighted Sum. The paper describes four different types of threat agents - Thief and Owner, Terrorist, Organized Crime and Mechanic, Hacktivist, and Foreign Government - and their motivations, capabilities, and window of opportunity for performing an attack. The paper provides an example of how the proposed attack feasibility rating concept can be applied to threat scenarios derived from the use cases and also compares it with other rating approaches such as attack-potential-based approaches, Common Vulnerability Scoring System (CVSS) based approaches, and attack vector-based approaches. The proposed attack feasibility rating concept is based on the attack-potential approach due to the complex nature of attacks against electric vehicles.
Other approaches include the CVSS and attack vectors. They conclude that attack-potential-based approaches have high flexibility but high complexity, CVSS-based approaches are easier to handle but have lower flexibility, and attack vector-based approaches are simpler but less suited for automotive applications.

These papers make a solid foundation for this proposed thesis by providing a to automatically find attack paths in one given architecture. However, they lack the ability to evaluate the given architecture by focusing only on one aspect of the architecture, namely the attack paths, rendering it once again a manual process for the security testers. This thesis aims to fill the gap in the existing research by providing a tool that combines the attack path analysis and architecture evaluation, and compare the architectures themselves.

3 PROBLEM STATEMENT

3.1 PROJECT TYPE

In this bachelor thesis, I will be conducting attack path analyses on different internal vehicle network architectures. I will then compare them based on which provides more security by introducing and using a tool that can be used to automate the evaluation of vehicular network security based on Attack Path feasibility. This bachelor thesis is done in cooperation with Mercedes-Benz Tech Innovation.

First, I will create multiple vehicular network architecture diagrams (A1).
Second, I will write a script-based tool that automizes the evaluation of said architectures (F1, F2, F3, F4, F5).
Every architecture consists of ECUs (nodes) and bus systems (edges) connecting the ECUs, as well as interfaces, each of which has an attack feasibility rating. The feasibilities are scaled based on information provided by security experts.

Next, I will decide on a criteria (A2), how to rate the different topologies. To decide on a criteria, I will conduct a survey with security architects and pentesters at MBTI, in which they will rate the different architectures based on their experience and knowledge.
The survey will be used to calibrate my own criteria.

Finally, I will compare the architectures to conclude which architecture offers the most security (A3).

The thesis will be both exploratory and implementory in nature, with a focus on the latter.

3.2 THESIS QUESTIONS

The questions this thesis will answer include:

- How can different E/E architectures be rated based on attack paths?
- How secure is the given vehicular network architecture?
- What architectural approach makes a network safer than others?
  - How do small changes in network positioning affect the network security overall?
  - How do simple and more branched out networks compare in terms of security?

4 TOOL DESIGN AND IMPLEMENTATION

4.1 PURPOSE

The tool's purpose is to help security architects quickly evaluate thegiven vehicular network architecture based on their Attack Path feasibility.

4.2 (FUNCTIONAL) REQUIREMENTS

(Functional) Requirements include but are not limited to:

General:

- A1 Multiple different vehicular network architecture diagrams as files of a certain datatype
- A2 A criteria to evaluate the architectures and a survey to calibrate the criteria
- A3 Compare architectures with one another and conclude which architecture offers the most security

Tool:

- F1: The tool will take files as input and parse them to a convenient datatype: One file contains the network diagram of the vehicle. The network diagram consists of ECUs (nodes), bus systems (edges) connecting the ECUs, and interfaces. In a separate configuration file, each ECU and each bus system will have an attack feasibility rating. It also specifies the entry points (ECUs and interfaces) and targets (ECUs).
- F2: A graph is created with the parsed data. The ratings in the configuration files are applied to the graph.
- F3: Next, an algorithm will find the most feasible attack path from each entry to each target based on the ratings. The algorithm can also be changed in the script.
- F4: The results are then output to a table containing the feasibility of each entry point to each target point together with the most feasible attack path.
- F5: Based on the table, the overall security of the network architecture is evaluated using the criteria (see 3.1 and A2).

4.3 OTHER REQUIREMENTS

Other requirements such as hardware, software, or non-functional requirements include, but are not limited to:

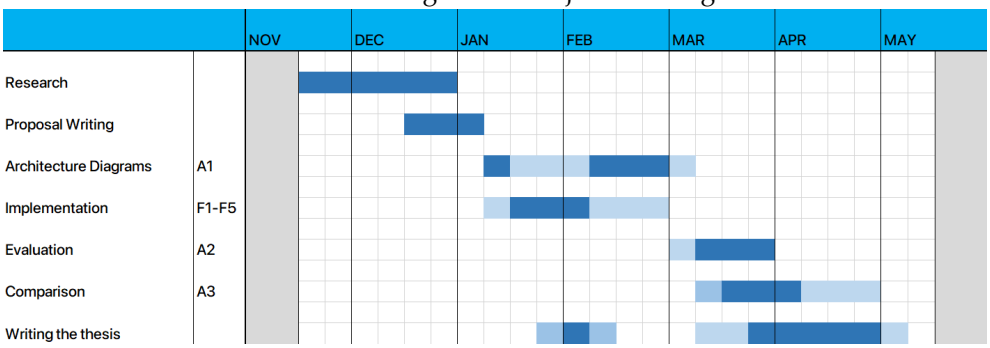
- N1: The tool must be script based

- N2: The programming language is Python (version not yet specified)
- N3: A virtual machine with a Linux distribution (distribution not yet specified)
- N4: Appropriate libraries for the tool
- N5: An IDE like PyCharm (recommended) or a text editor to change the script
- N6: A Git repository where the code and thesis will be stored

5 PROJECT SCHEDULE

5.1 THESIS TIMELINE

The coarse schedule looks as following and is subject to change:



5.2 CONTINGENCY PLANS

The project is subject to change.
New insights gained during the project might affect some aspects of tool implementation (4) or diagrams (4.2) This may include implementation problems, such as library or datatype incompatibilities, that might arise while working on the tool. It can be addressed using other libraries, datatypes, or by changing the implementation.

ACRONYMS

AUTOSAR AUTomotive Open System ARchitecture. 4

CAN Controller Area Network. 1, 5
CVSS Common Vulnerability Scoring System. 6

E/E Electronic/Electrical. 1, 3
ECU Electronic Control Unit. 1, 4, 7, 8
EFSM Extended Finite State Machine. 5
EVITA E-safety vehicle intrusion protected applications. 4

HEAVENS HEAling Vulnerabilities to Enhance Software, Security, and Safety. 4

OBD On-board diagnostics. 5

STRIDE Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege. 4

TARA Threat Analysis and Risk Assessment. 1, 4

GLOSSARY

Attack Path As defined in [2], an attack path is a set of deliberate actions that an attacker takes to realize a threat scenario, a potential cause of compromise of cybersecurity properties of one or more assets in order to realize a damage scenario. . 1, 5, 7, 8

BIBLIOGRAPHY

[1] International Organization for Standardization, *Road vehicles – functional safety*, Geneva, Switzerland: International Organization for Standardization, 2018. [Online]. Available: https://www.iso.org/standard/64539.html.

[2] International Organization for Standardization, *Road vehicles – cybersecurity engineering – guideline and general aspects*, Geneva, Switzerland: International Organization for Standardization, 2020. [Online]. Available: https://www.iso.org/standard/73547.html.

[3] SAE International, *Cybersecurity guidebook for cyber-physical vehicle systems (sae j3061)*, Warrendale, Pennsylvania, USA: SAE International, 2018. [Online]. Available: https://www.sae.org/standards/content/j3061_201801/.

[4] AUTOSAR Development Partnership, *Autosar 4.2.2 specification*, Munich, Germany: AUTOSAR Development Partnership, 2019. [Online]. Available: https://www.autosar.org/standards/standard-downloads/.

[5] National Institute of Standards and Technology, *Threat analysis risk assessment (tara)*, Gaithersburg, Maryland, USA: National Institute of Standards and Technology, 2011. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-30/rev-1/final.

[6] European Union's Horizon 2020 research and innovation programme, *Heavens: Healing vulnerabilities to enhance software security and safety*, Brussels, Belgium: European Union's Horizon 2020 research and innovation programme, 2019. [Online]. Available: https://cordis.europa.eu/project/id/866041.

[7] National Institute of Standards and Technology, *Evaluating the vulnerability of information technology assets (evita)*, Gaithersburg, Maryland, USA: National Institute of Standards and Technology, 2003. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-53/rev-4/final.

[8] F. Sommer, R. Kriesten, and F. Kargl, "Model-based security testing of vehicle networks," in *2021 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2021, pp. 685–691. DOI: 10.1109/CSCI54926.2021.00179.

[9] F. Sommer and R. Kriesten, "Attack path generation based on attack and penetration testing knowledge," 2022.

[10] J. Dürrwang, F. Sommer, and R. Kriesten, "Automation in automotive security by using attacker privileges," 2021. [Online]. Available: https://hss-opus.ub.ruhr-uni-bochum.de/opus4/frontdoor/index/index/year/2021/docId/8357.

[11] D. Zelle, C. Plappert, R. Rieke, D. Scheuermann, and C. Krauß, "Threatsurf: A method for automated threat surface assessment in automotive cybersecurity engineering," *Microprocessors and Microsystems*, vol. 90, p. 104 461, 2022, ISSN: 0141-9331. DOI: https://doi.org/10.1016/j.micpro.2022.104461. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0141933122000321.