Ontology-Based Model for Automotive Security Verification and Validation

Abdelkader Magdy Shaaban
Austrian Institute of Technology
Center for Digital Safety & Security
Vienna, Austria
abdelkader.shaaban@ait.ac.at

Christoph Schmittner
Austrian Institute of Technology
Center for Digital Safety & Security
Vienna, Austria
christoph.schmittner@ait.ac.at

Thomas Gruber
Austrian Institute of Technology
Center for Digital Safety & Security
Vienna, Austria
thomas.gruber@ait.ac.at

A. Baith Mohamed
University of Vienna
Faculty of Computer Science
Vienna, Austria
abdel.baes.mohamed@univie.ac.at

Gerald Quirchmayr
University of Vienna
Faculty of Computer Science
Vienna, Austria
gerald.quirchmayr@univie.ac.at

Erich Schikuta
University of Vienna
Faculty of Computer Science
Vienna, Austria
erich.schikuta@univie.ac.at

ABSTRACT

Modern automobiles are considered semi-autonomous vehicles regarding new adaptive technologies. New cars consist of a vast number of electronic units for managing and controlling the functional safety in a vehicle. In the vehicular industry, safety and security are considered two sides for the same coin. Therefore, improving functional safety in the vehicular industry is essential to protect the vehicle from different attack scenarios. This work introduces an ontology-based model for security verification and validation in the vehicular domain. The model performs a series of logical quires and inference rules to ensure that the security requirements are fulfilled. It endeavors to enhance the current security state of a vehicle by selecting additional security requirements that can handle existence security weaknesses and meet the actual security goal.

CCS CONCEPTS

• **Security and privacy** → *Systems security*;

KEYWORDS

Ontology, Automotive, Security Requirements, Threats, Protection Profile, Verification and Validation.

ACM Reference format:
Abdelkader Magdy Shaaban, Christoph Schmittner, Thomas Gruber, A. Baith Mohamed, Gerald Quirchmayr, and Erich Schikuta. 2019. Ontology-Based Model for Automotive Security Verification and Validation. In *Proceedings of The 21st International Conference on Information Integration and Web-based Applications & Services, Munich, Germany, December 2–4, 2019 (iiWAS2019),* 8 pages.
https://doi.org/10.1145/3366030.3366070

1 INTRODUCTION

The vehicular industry is rapidly evolved from mechanical units working on gears and shafts to electronic components interacting using different communication protocols. The modern vehicles combine a considerable number of interconnected units as Sensors, Electronic Control Units (ECUs), Buses, Actuators, and other electronic elements for monitoring and controlling the state of the vehicle [18]. Next-generation vehicles will include 20+ computers with storage sizes range from 8GB to 256GB [21]; wherever Voyager 1 and Voyager 2 have 69.63 kilobytes of memory for each [22]. Furthermore, modern vehicles have more powerful processing capabilities than the former space probs. The high-end car has over 100 million lines of code, and it is expected that the number would continue to grow shortly. Such codes are implemented for various control applications over numerous functionalities like safety-critical functions, driver-assistance, and others. The software operates on hundreds of programmable ECUs that interact via several types of communication protocols and buses (i.e., Controller Area Network (CAN bus), FlexRay, and Ethernet) [6].

A research group from the University of Washington and the University of California San Diego has proved that it is possible for a code stored in any electronic unit to control critical components in a vehicle such as the brakes system. They have demonstrated that attackers can inject malicious code with physical access to the vehicle or even remotely using different wireless communication methods. This illustrates the real threat is not the accidental failure of any components in the vehicle, but the consequence of malicious code on the functional vehicular safety [18]. Accordingly, cybersecurity needs to be a part of the designing phases of the vehicular industry. Cybersecurity in the vehicular domain plays an integral role because it is responsible for protecting components and software that are managing the functional safety in a vehicle from different attack scenarios (i.e., unauthorized access, information infiltration, man-in-the-middle, or others) [23]. Moreover, to improve the functional safety in the current and future vehicular industry, it is essential to develop requirements for vehicular components to assure their reliability and security [31]. However, the diversity in communication protocols and heterogeneity of electronic parts in the vehicle lead to an increase in the abundance of security vulnerabilities. Further, the process of security verification

iiWAS2019, December 2–4, 2019, Munich, Germany

ACM ISBN 978-1-4503-7179-7/19/12…$15.00
https://doi.org/10.1145/3366030.3366070

and validation (V&V) will be more complicated because this process must be aware of all vehicular components, potential threats, and related security requirements, which is considered a challenging process.

This research proposes an ontology-based model for vehicular security V&V. The model uses the ontology approach to represent the vehicular components, threats, vulnerabilities, and security requirements. The security requirements are defined in terms of a group of documents that are called protection profiles (PP) [3]. The PP describes the security considerations for a Target of Evaluation (ToE) according to Common Criteria (CC). The ToE is a conceptual explanation of a system or system unit for a particular usage that is subjected to the evaluation. The ontology helps to create a complete overview of vehicle ToEs, related threats, and selected security requirements that are to be used in the validation and verification process. The model supports logical queries and inference rules to determine whether or not the selected security requirements are completely fulfilled. Additionally, the model improves the current security level of a vehicle by selecting additional security requirements from several PPs. This leads to handle some existence security gaps and reach to the actual security goal that is needed to be achieved.

This paper is organized as follows; a short discussion about the existing research contributions in automotive security is presented in Section 2. Section 3 introduces the main concept of this work to describe the building blocks of the proposed ontology-based security V&V Model. The model is applied to a modern vehicle case-study to investigate the potential threats and related security requirements. Then the model verifies and validates the selected security requirements as described in Section 4. Section 5 demonstrates the influence of ontologies in the V&V process to manage a considerable amount of security requirements. Finally, the paper ends with a summary, conclusion and presents future work.

2 RELATED WORK

Any such device connects to the internet is exposed to be attacked by different ways of malicious activities, as the same as modern vehicles, which are considered as a complex system contains a vast amount of interconnecting objects [16]. Furthermore, vehicular cybersecurity is becoming one of the primary research topics in the automotive industry. The security engineering process in the vehicular domain contains sequence stages of activities that need to be conducted with the automotive development lifecycle. Figure 1 depicts the main activities of the security development phases in the automotive domain.

The security engineering process is ensured that the vehicle development phases can [4]:

- "Risk Analysis:" define the exact security vulnerabilities and potential threats in a vehicle,
- "Risk Evaluation:" correctly assess the risk on a vehicle,
- "Risk Treatment:" identify the most suitable security requirements able to address the security breaches in a vehicle and reduce the overall risk,
- "Security Assurance:" verify and validate that the security requirements meet the actual demanded security level.

2.1 Risk Analysis

One single vulnerable unit in a vehicle could make all components and vehicles to be exposed to a higher degree of surface attacks. Furthermore, it is essential to understand the exact security weaknesses in a vehicle at the early stages of the security engineering process because once the vehicle is built is becomes more difficult to add security.

2.1.1 *Target Component.*

It is essential to identify all components in a vehicle, to be used further in the other phases of the cybersecurity management process. This phase determines the security properties (Security Mitigations) of the defined components (such as Secure Boot, Authentication, Encryption, others). These properties are essential in the threat and vulnerability analysis processes and for selecting the most relevant security requirements to address the identified security weaknesses in a vehicle.

2.1.2 *Threat Analysis.*

Threat analysis is an activity that identifies potential negative actions that affect the security mechanism in the vehicles [4]. Ref. [9] discusses an overview of the available solutions for the threat modeling process. The threats and risk assessment techniques are mentioned in several research topics. Ref. [15] reviews the available techniques in the vehicular sector of threats and risk evaluation; then, it presented an approach to classify security threats. [14] demonstrated that threat modeling, using existing tools, can be a helpful and effective analysis method for the automotive security engineering process in various stages in the automotive development lifecycle.

In addition, the authors have introduced a threat modeling approach in the automotive domain that can be simply integrated with the automotive security engineering process. This approach is called Threat Management Tool - ThreatGet [5], [9]. ThreatGet identifies, detects, and understands potential threats in the early stages of the design phase of vehicles. In the analysis process, the tool uses the security properties of the components, which are incorporated with the target elements. Besides, it uses a threat catalog that contains a wide range of potential threats in the vehicular domain to define the exact potential threats. The following source documents were used to develop the threat catalog:

- Threat Modeling for Automotive Security Analysis [14].
- Connected Cars - Threats, Vulnerabilities and Their Impact [34].
- Threat Landscape and Good Practice Guide for Internet Infrastructure [13].
- A survey of Remote Automotive Attack Surfaces [19].

The threat catalog has been implemented based on grammar as a formal structure using constant rules. The threat tool classifies the potential threats into six main groups according to the STRIDE model (i.e., Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service (DoS), and Elevation of Privilege) [11].

2.1.3 *Vulnerability Identification.*

The vulnerability analysis is the process of exploring, defining, identifying, and prioritizing vulnerabilities or security weaknesses. Meanwhile, security mechanisms need to be used to avoid those threats that exploit existing

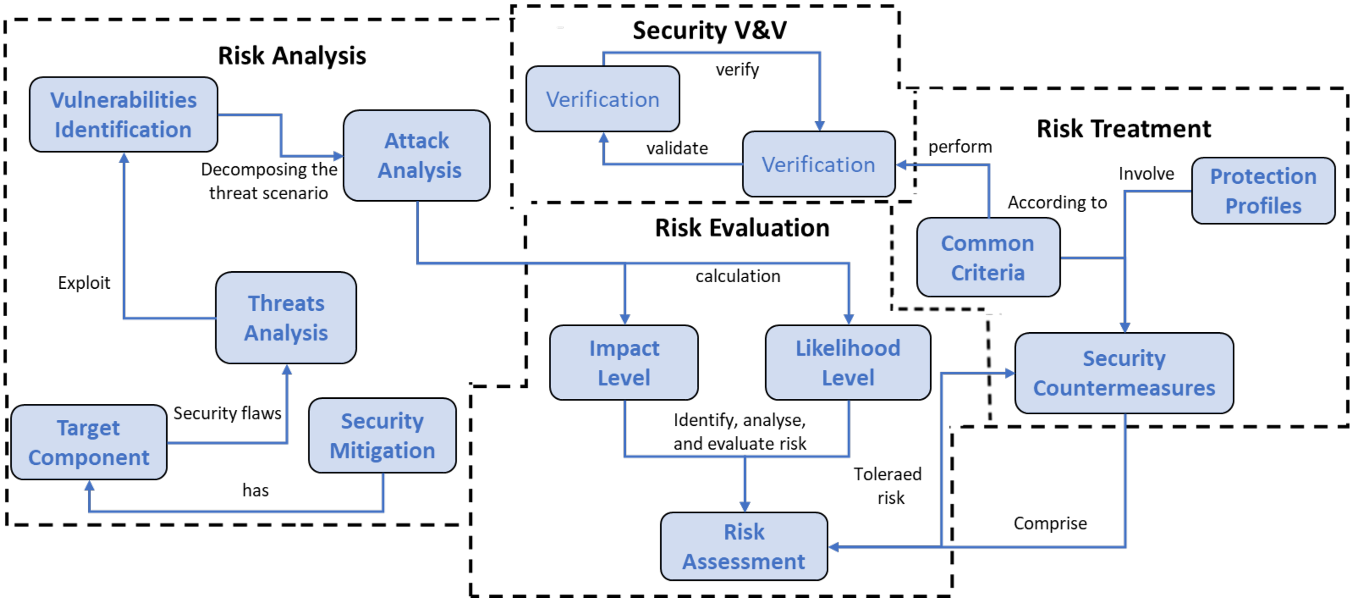

Figure 1: Security development activities in the vehicular industry

vulnerabilities in system units. The security mechanism is composed of several types of defensive actions such as detective, preventive, corrective, recovery, or response [20]. A brief overview of security vulnerabilities and existing vulnerability databases in the automotive domain is discussed in [32]. Vulnerabilities can be found at hardware, software, or network level, an overview of a possible classification of vulnerabilities has been presented in [29].

2.2 Attack Analysis

The attack analysis aims to define the relationships between the detected threats and the discovered vulnerabilities. This analysis tries to trace a massive number of security threats that may exploit vulnerabilities to attack a vehicle; this process is called attack paths. Also, the attack paths process aims to collect and derive information about the paths that are used by the attacker to attack the vehicle. This information could also assist in the security testing process [32].

2.3 Risk Evaluation and Treatment

Risk evaluation or risk assessment process is a systematic approach of identifying and analyzing the hazards (i.e., safety) or threats (i.e., security) and estimating a level of risk severity for each hazard or threat [25]. This activity is based on the parameters of impact and likelihood, which are used to evaluate the specific risk level. On the first hand, it is essential to ensure that different types of impacts do not damage the vehicle or cause other accident scenarios. Ref. [30] described four levels of impacts in the automotive domain:

- causes immediate damage to the environment or human lives (safety),
- causes the loss of control over personal information (privacy),
- causes financial damage (finance),
- negatively impacts the operation and traffic flow (operation).

The evaluation of the likelihood considers the significant factor in the risk assessment process. The likelihood values represent how straightforward it is to exploit security weakness to attack a vehicle. Four different perspectives are proposed to evaluate the likelihood (i.e., attacker capabilities, ease of gaining information, accessibility of the system, and required equipment for an attack) [30]. Later, the level of severity of each of the detected potential threats is evaluated based on parameter values of the likelihood and impact level.

The risk assessment process uses several risk methods for evaluating the vehicular risk level based on the parameter values of the likelihood and impact. The following formula is one of the most common risk assessment method:

$$Risk = Threat * Vulnerability * Consequence \qquad (1)$$

where:

$$\begin{aligned} Threat * Vulnerability &= \text{Likelihood} \\ Consequence &= \text{Impact} \end{aligned}$$

The next steps are to address the unacceptable risk with applicable security requirements that reduce the risk severity level. Figure 2 depicts an example of the evaluated risks of the detected threats, as pointed on the graph. In this example, it is expected that the Tolerable Value (TV) or the risk acceptance threshold is four. The TV represents a security threshold; all values above the TV need to be addressed by the suitable security requirements to mitigate the risk.

For example, the threat (T1) on the graph is classified as an extreme severity level. The value of the Security Target (ST) is set during the concept phase, to define the specific security goal. Therefore, the security requirement(s) is/are used to mitigate the risk to an acceptable level. The resulting state after applying security

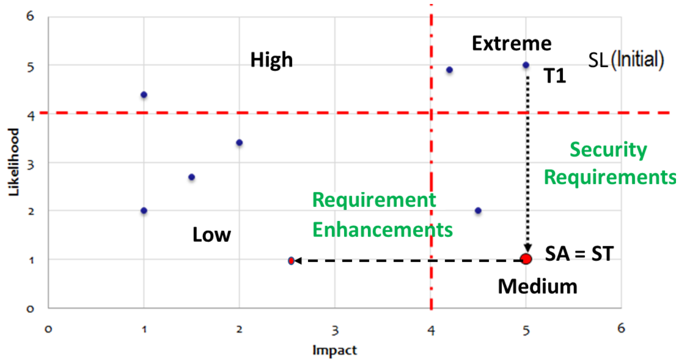

Figure 2: Risk Mitigation Process [4]

requirements is called the Security Achieved (SA). This process completes only if SA = ST; otherwise, other security requirements (Requirement Enhancement (RE)) have to be applied to reduce the risk further to an acceptable level. [2] discussed in detail about security target, security achieved, and requirement enhancement.

In addition, the ontology approach is described in several research topics in cybersecurity. A CC Ontology tool is introduced as an ontology-based method to assist the evaluator at the certification process [7]. [33] introduced a security ontology for security requirements engineering that supports in the elicitation of security requirements. A security ontology is presented in [8] to provide a stable base for an applicable and holistic IT-security approach for small and medium-sized enterprises (SMEs), allowing low-cost risk management and threat analysis.

2.4 Verification & Validation

During and after the vehicular security engineering process, the vehicle must be checked to ensure that it is implemented according to the highest degree of protection level. Ref. [12] introduced a model-based security testing in the automotive industry; it discussed that the verification process of security requirements is integrated late in the development stages, where both time and budget are very restricting circumstances.

The main contribution of this work is introducing an ontology-based model for the V&V process. This concept comes after all previously discussed phases of the security development lifecycle.

3 THE BUILDING BLOCKS OF ONTOLOGY SECURITY V&V MODEL

The verification and validation processes are considered an integral part of this work. This section discusses the building blocks of the proposed ontology-based model to verify and validate the security requirements against potential threats in the vehicular domain. The core of this model is OnSecta - Ontology Security Testing Algorithm. The OnSecta performs security verification and validation according to the current security status, and the actual security goal needs to be achieved. Figure 3 describes the building blocks of the proposed ontology model.

Reading Data: The block accepts the details of components, potential threats, and selected security requirements. Then it generates a comprehensive ontological overview of threats and related security requirements. This ontology overview is called "Ontology Outlook." The Ontology Outlook has two hierarchies:

- **Threats:** it is an ontological representation of all identified potential threats.
- **Security Requirements:** it is a descriptive semantic representation of security requirements that are correlated to certain PP(s) for addressing potential vehicle threats.

The structure of the generated Ontology Outlook is described as semantic annotations (triples) in the form of (subject, predicate, and object); where the subject is the detected potential threat, the predicate is an object property assertion between threat and security requirements, and the object is security requirement. For example, a threat (T) can be addressed by a related security requirement (SR); so that it will be described as:

$$T \xrightarrow{addressedBy} SR$$

The predicate of the generated ontology is expressed as links between the threats hierarchical nodes and the security requirements nodes. That represents the selected security requirements can address one or more potential threat(s).

Verifier: The verifier part is one of the main blocks of the OnSecta design. That acts a peer review analysis to verify every single node in the Ontology Outlook to check the formal correctness or integrity, of a specific threat, that is addressed by security requirement(s). The verification process verifies if the specifications of the security requirements meet at the actual security level, which needs to be achieved to address a specific level of risk severity. For example, threat severity level plays an integral part in the risk treatment and V&V processes, because a threat with high severity level needs to be addressed with at least SL3 security requirement(s), as will be described in Section 4.

The verifier uses SPARQL query language to review the properties of the selected security requirement(s) and match the severity level of threats. The SPARQL language is applied to the Ontology Outlook to perform queries across different data sources(threats and security requirements) [26]. These queries are used to ensure that a vehicle is being developed based on standard security requirements, according to CCs. Additionally, to assures, the compliance of ToEs with PP meet the exact ST.

Validator: The validator part aims to investigate if the selected security requirements meet the ST. The validator block is considered as a rule-based approach consists of a set of "if-then" clauses to check which of the security requirements are validated or not. It uses SQWRL language (Semantic Query-Enhanced Web Rule Language) [24] that presents SQL-like operators for obtaining information from ontologies. The validator method construct SQWRL queries according to CC of security requirements are stored in an Ontology Knowledge Base (OKB). The CC is defined in a separate file "Requirement Properties" that represent the specific properties of threats and the related security requirements. The "Rule Generator" unit creates queries; then the "Query Engine" execute these queries to find security requirements able to address specific threats

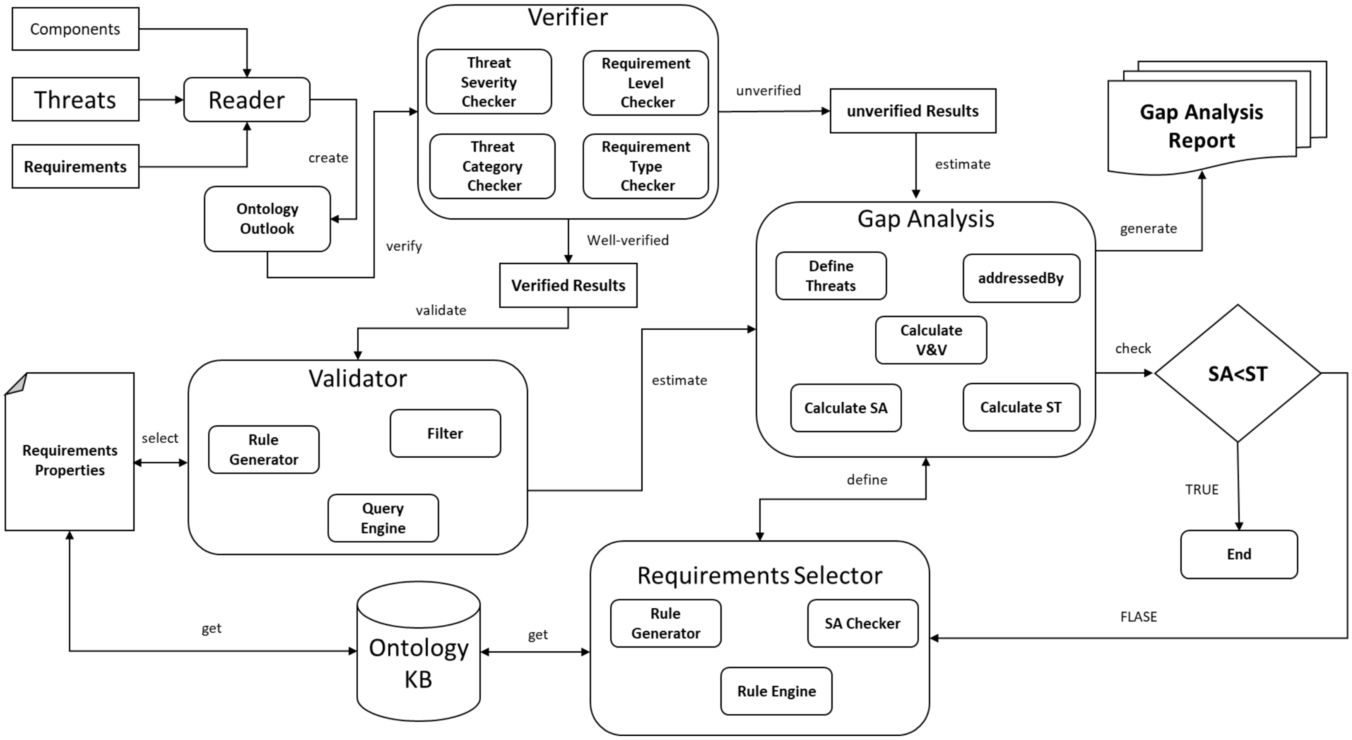

Figure 3: The building blocks of the V&V model

according to the CC. The "Filter" block selects the most suitable security requirements according to particular matching properties with threats.

Gap Analysis: The Gap Analysis method is applied to asses the differences in the SA before and after verifying and validating the selected security requirements. This estimates whether the selected security requirements meet the actual ST, and defines how to improve the current security state. This method calculates the values of SA and ST; then it generates reports that describe a complete view for describing the impact of the applied selected security requirements to the detected potential threats.

Requirements Selector. The selector method uses the results of the gap analysis and performs a series of inference rules to select new security requirements from other PPs in the OKB to reach the actual ST. The "Rule Generator" generates logical rules that are typically conditional if-then clauses. The Semantic Web Rule Language (SWRL) [17] is applied to represent knowledge that select new security requirements from the OKB according to particular CC to address threats and achieve the required ST. The generated rules are applied to the "Rule Engine" to infer the logical consequences of the defined threats and security requirements properties. Then, it suggests a new set of security requirements suitable according to specific CC to address particular security weaknesses. The process continues until the SA = ST; otherwise, the OnSecta applies different rules using other PPs until the equation is satisfied.

4 CASE-STUDY: MODERN AUTOMOBILES

Automobiles are no longer mechanical units; the modern vehicles contain a massive number of interconnected electronic components networked together for controlling and monitoring the state of the vehicle. Modern vehicles consist of around 50 connected Electronic Control Units (ECUs) [18]. The increase in connectivity and interaction between multiple devices leads to the rise of new hazards. Figure 4 shows a simple design of a modern vehicle that contains numerous interconnected units.

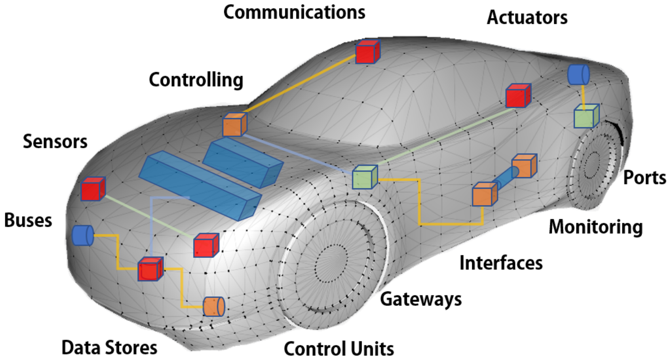

Figure 4: Interconnected units in a modern vehicle

This case study aims to investigate the security weaknesses and the suitable security requirements as an example of interconnected units in a modern vehicle. This work uses the IEC 62443 standard series to select applicable security requirements to address potential threats in the automotive domain and to keep the risk low. According to the IEC 62443 security standard [2], the associated four Security Levels (SLs) are described as [10]:

- **SL1:** unintended,
- **SL2:** low resources, generic skills and low motivation (i.e., simple means),
- **SL3:** moderate resources, moderate motivation (i.e., moderate means),
- **SL4:** extended resources and high motivation (i.e., sophisticated means).

The security requirements in IEC 62443 standard series are defined according to different seven foundational requirements (FRs) [2], [1].

- Identification and Authentication Control (IAC),
- Use Control (UC),
- System Integrity (SI),
- Data Confidentiality (DC),
- Restricted Data Flow (RDF),
- Timely Response to Events (TRE), and
- Resource Availability (RA).

Figure 5 describes communication flow among multiple internal components which are used to represent modern automobile. The example has a "Sensor" unit that receives data from the outer environment. The sensor sends the obtained data to the "Sensor Control Unit" for manipulating the input data. Then the "Telematics" unit uses these data to control the vehicle tracking system. It interacts with the "Advanced Driver Assistance System" to send signals to the "Brakes" to control the vehicle acceleration according to different traffic circumstances. The car interacts with the outer environment through the V2X-getaway for activating safety functionalities in the vehicle.

The ThreatGet tool is applied to the identified components for detecting potential threats in this model. Additionally, the authors developed a Model-based Security Requirement Management Tool (MORETO) for managing a vast number of different security requirements. Therefore, MORETO is used in this work to manage the IEC 62443 security standard to address security weaknesses. Afterward, OnSecta will be applied to the results (threats and security requirements) to perform the verification and validation process.

In this example, ThreatGet detects 56 potential threats that are classified according to the STRIDE model. Figure 6 illustrates the rate of the detected threats according to the STRIDE model. Besides, the tool evaluates these threats to estimate the overall risk. Table 1 shows the outcomes of the severity assessment process.

Table 1: The results of the severity assessment process

Severity	Number
Extreme	7
High	13
Medium	17
Low	19

There are seven threats categorized as extreme, and 13 threats are evaluated as a high degree of severity. The ThreatGet estimates 17 and 19 threats as medium and low degree of severities, respectively. Subsequently, the MORETO tool is applied to this example to manage the stored security requirements and to facilitate selecting the security requirements that are necessitated to address the detected potential threats to minimize the overall risk.

Consequently, OnSecta is applied to validate and verify the selected security requirements. OnSecta creates multiple classes, subclasses, individuals, properties, and annotations of all detected potential threats and selected security requirements. Then it generates the Ontology Outlook to create a comprehensive overview of all threats and security requirements, as addressed in Section 3. Figure 7 illustrates the structure of the Ontology Outlook; this structure consists of three main parts:

- **Potential Threats (left-side):** hierarchy of all the vehicular components that are used in this case study. Besides, the identified potential threats that are detected by the ThreatGet tool.
- **The Security-Requirements(right-side):** represents a hierarchy of all security requirements that are selected to handle the detected potential threats.
- **The Mapping Between the Two Ontologies (middle):** these links represent specific security requirements are address one or more potential threat(s). The ontology on the left-side (threats) can be linked to a subtree on the right ontology (security requirements) that indicates a set of similar security requirements that can address a potential threat for handling related security issue [27], [28].

According to the OnSecta building blocks, the verifier and validator blocks use the generated Ontology Outlook to determine the value of the SA and ST. In this work, the authors propose that the threats of extreme severities are unacceptable. The means the extreme threats are needed to be addressed to particular security requirements according to specific properties to reduce the overall risk of the vehicle. For example, the number of extreme threats according to Table 1 equal to seven, so the value of ST = 7. In this case-study, OnSecta has to verify and validated the selected security requirement to assure that the value of SA = ST.

Based on the outcomes of the verifier and the validator, OnSecta creates the gap analysis to demonstrate which of the selected security requirements are fulfilled. OnSecta observes validated security requirements entirely handle three extreme threats according to precise security properties. Table 2 represents the gap analysis results with all details of the investigated threats against associated security requirements.

The gap analysis table contains all details about the threats that classified as extreme severity. The result of the V&V process is an integral part of this table to determine the current and target security states. For example, Threat21 is addressed by two security requirements (UC11 and UC11.2). The OnSecta verifies and validates these requirements based on the specific properties; the first security requirement (UC11) is validated, where the second one (UC11.2) is invalidated. Furthermore, the overall evaluation of that threat is FALSE. The target level which needs to be reached is four

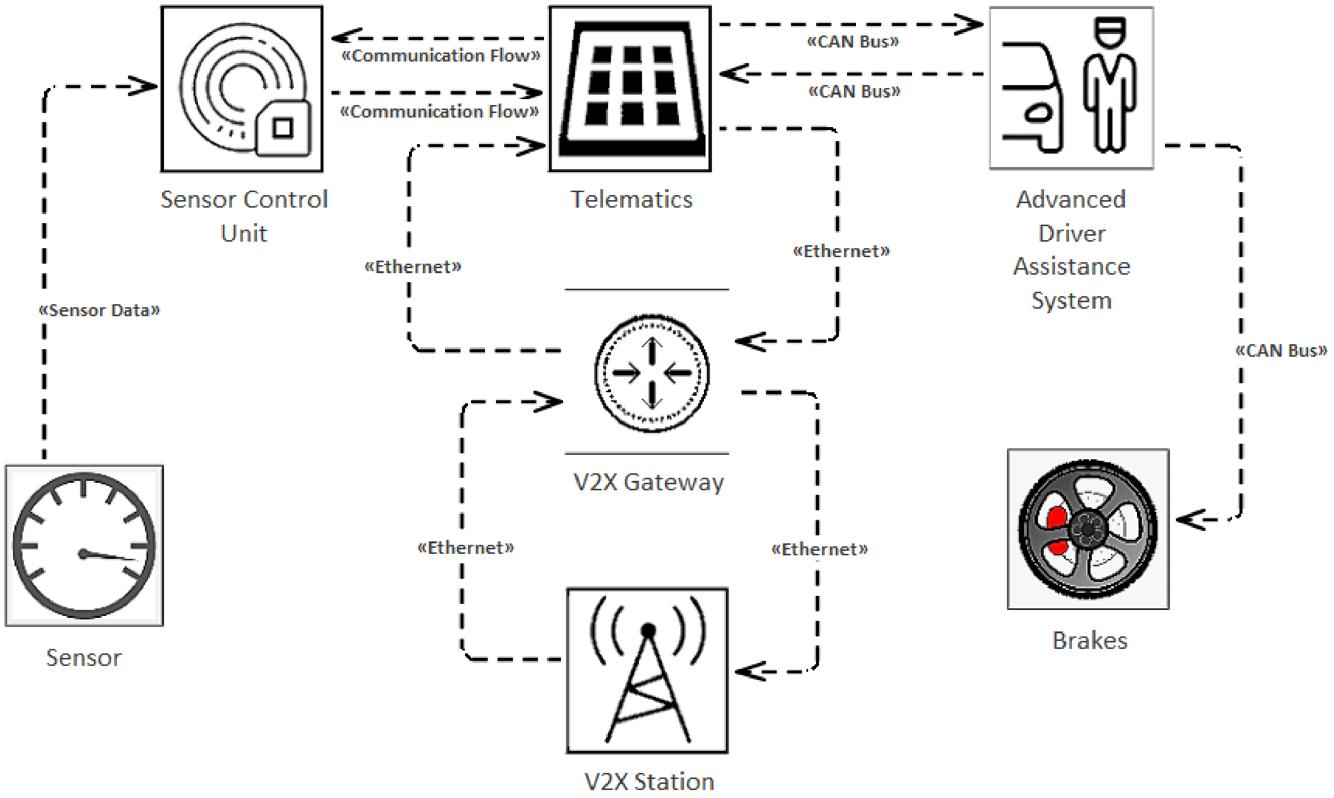

Figure 5: Communication flow among interconnected units in a modern vehicle

Table 2: The results of gap analysis process

#	Details											V&V	Evaluation
	Threat ID	Threat Name	Category	Severity	Addressed by						Target		
					Requirements	SL1	SL2	SL3	SL4	Current			
1	321	Threat21	Repudiation	Extreme	UC11		x	x	x	2	4	TRUE	FALSE
					UC11.2				x			FALSE	
2	354	Threat54	Spoofing	Extreme	UC1	x	x	x	x	1	4	FALSE	FALSE
3	350	Threat50	Spoofing	Extreme	IAC5	x	x	x	x	2	4	TRUE	FALSE
					IAC5.1			x	x			FALSE	
4	355	Threat55	Spoofing	Extreme	SI8		x	x	x	4	4	TRUE	TRUE
					IAC2		x	x	x			TRUE	
5	301	Threat1	Denial of Service	Extreme	IAC9		x	x	x	1	4	FALSE	FALSE
6	328	Threat28	Spoofing	Extreme	IAC2		x	x	x	4	4	TRUE	TRUE
					SI8		x	x	x			TRUE	
					IAC5	x	x	x	x			TRUE	
					IAC2.1			x	x			TRUE	
7	319	Threat19	Information Disclosure	Extreme	IAC11	x	x	x	x	4	4	TRUE	TRUE
					IAC8		x	x	x			TRUE	

because the selected threat is classified as extreme severity level. The Current level value is evaluated according to (2).

$$CurrentState = \sum_{index=1}^{n} \frac{(V\&V * SL_{max})}{n} \tag{2}$$

where:

n = number of addressedBy (security requirements)
$V\&V$ = the result of V&V (TRUE = 1 and FALSE = 0)
$SLmax$ = is the maximum security level

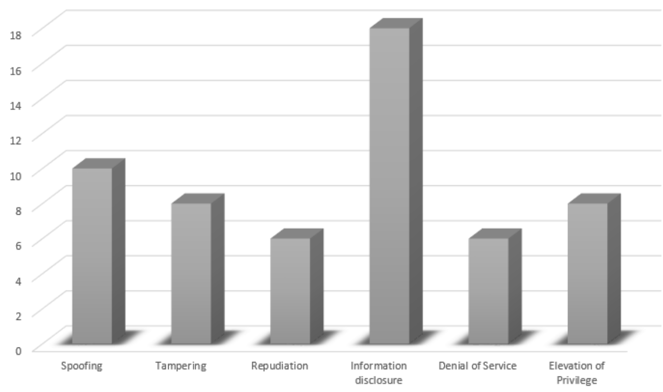

Figure 6: Classification of the detected potential threats according to the STRIDE model

This equation is applied to all extreme threats to determine the current security level. Afterward, OnSecta estimates the overall current security state (SA) by defining the average value of all extreme threats. The last step in the gap analysis process is to define the SA of the example given. The value of the SA is calculated according to the number of extreme are verified and validated by OnSecta. Due to the gap analysis results, the SA = 3. Figure 8 illustrates a graphical comparison between the current security status (blue) and the target security level (orange) that needs to be achieved.

OnSecta applies a series of rules to specify additional security requirements to help in covering the security gaps, as defined in Table 2 . OnSecta uses the OKB and the "Requirements Properties" as described in Section 3 to select additional security requirements according to particular CC (properties of threats and security requirements). These new security requirements are used to manipulate existing security weaknesses and assure compliance with security requirements to meet the ST. The verification and validation processes are applied whenever the rules choose new security requirements. The verified and validated security requirements will substitute the incorrect ones. Then, the SA is estimated once again until the SA = ST.

5 MODEL EVALUATION

The OnSecta uses the OKB to select additional security requirements from one or more PPs that can handle the security weaknesses as described in the table of the gap analysis. The rules are created according to particular features of the security requirements and automotive units. Figure 9 illustrates the rate of the security requirements before applying the OnSecta rules.

The security requirements are defined according to the seven foundational requirements as previously described according to the IEC 62443 standard [2]. The OnSecta applies inference rules using the stored security requirements in the OKB to select extra security requirements to address the unhandled extreme potential threats. The rate of the inferred security requirements is illustrated in Figure 10.

By observing, there are two security requirements have been addressed the Threat50, according to the gap analysis (Table 2), the IAC5 is validated; where the IAC5.1 is not invalidated. However, the rules inferred new two security requirements that can handle this threat. Also, the rules can ultimately select distinctive security requirements that are more suitable to address a particular potential threat. For example, Threat21 is addressed by two security requirements as UC foundational requirement, as shown in Figure 9. However, the model finds another one security requirement of the same FR (i.e., UC) can also become proper to address the equivalent threat, as depicted in Figure 10. On the first hand, OnSecta aims to enhance the security level of a vehicle; that is the reason some of the well verified and validated security requirements are enhanced as is happened with Threat28 and Threat19, or stay the same without changes as Threat55. On the other hand, it tries to give the most optimum security requirements for addressing particular security weaknesses. From the cost point of view, selecting precise solutions are cost-effective than applying generic ones for tackling a particular problem. The ontology approach proves that it is a robust methodology to manage a vast amount of security requirements. The structure of the Ontology Outlook helps in reducing the query time complexity. The features of the ontology demonstrate that it is the most suitable methodology to be used in the vehicular industry to manage hundreds or thousands of security requirements and potential threats to tackle the challenges of the security verification and validation process in the automotive domain.

6 SUMMARY, CONCLUSIONS AND FUTURE WORK

The modern vehicles are consist of a massive number of interconnected units communicating through a network. Vehicular safety and security become two sides of the same coin. The relationship between safety and security is considered a directly proportional because any injected malicious code to any components or busses in a vehicle leads to damage or malfunction, which threaten the functional safety in a vehicle.

This work presented an ontology-based model for the security verification and validation process in the vehicular domain. The model verifies and validates security requirements in a vehicle to assure that these requirements are fulfilled. It creates a comprehensive ontology view of vehicle components, detected potential threats, and related security requirements to verify and validate the requirements against the potential threats. Then the model applies a sequence of logical queries to the ontology view to determine whether or not the security requirements are able to handle risks in a vehicle.

A modern vehicle example is used in this work as a case-study to investigate both potential threats and security requirements. The OnSecta model verifies and validates the selected security requirements, then calculates the value of the current security state (SA). The value of SA = 3 (validate and verify security requirements), where the security goal of this example was seven (ST = 7). The OnSecta model performs a series of inference rules to select additional security requirements from the OKB that can handle the security weaknesses in a vehicle. This process is repeated until the value

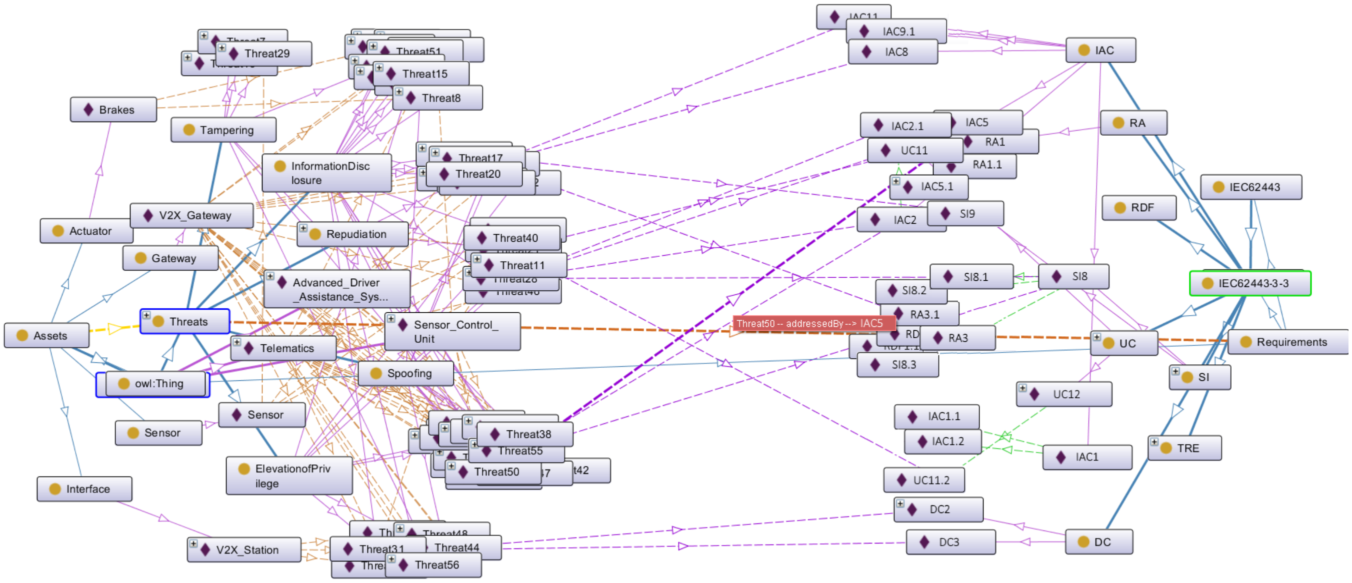

Figure 7: The generated ontology outlook of this modern automobiles case-study

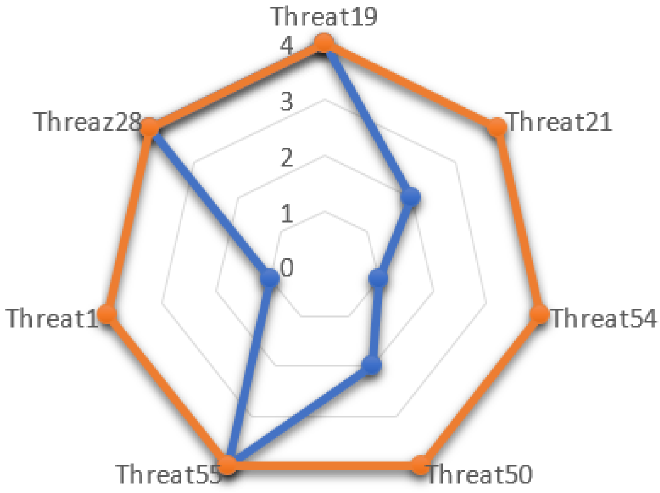

Figure 8: The current (blue) and target security (orange) levels

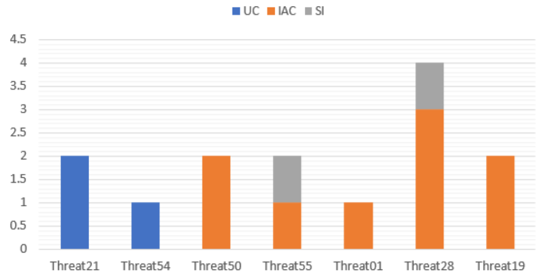

Figure 9: Number of the selected seucirty requirements before applying Rules

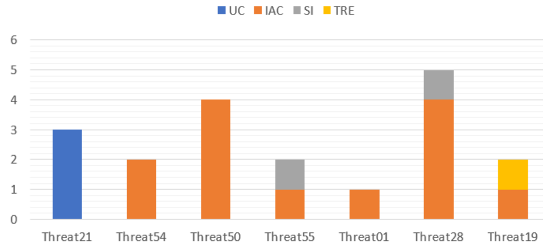

Figure 10: Number of the selected seucirty requirements after applying Rules

of SA = ST, which means the selected security requirements are wholly fulfilled.

This work is based on IEC 62443-3-3 standard [2] for demonstrating the first steps in our proposed concept. The future works aim to add additional security requirements to the ontology KB to ensure a wide range of security requirements and PPs with specific security requirements are completely fulfilled in the automotive domain. The partial list of ISO/IEC standards includes:

- IEC 62443-x-x: Industrial communication networks – network and system security.
- ISO 27000-series for Information Security Management System (ISMS).

Then, the future work will also include more technical aspects about the proposed model and a comparative investigation between the proposed method with other kinds of common methods in the related field to validate the advantage of the proposed method.

ACKNOWLEDGMENT

This work has received funding from the iDev40 project. This project is co-funded by the consortium members and ECSEL Joint Undertaking under grant agreement No 783163. The JU receives support from the European Union's Horizon 2020 research and innovation programme, national grants from Austria, Belgium, Germany, Italy, Spain and Romania as well as the European Structural and Investement Funds.

REFERENCES

[1] IEC 62443-3-1(TR): Industrial communication networks – network and system security – part 3-1: Security technologies for industrial automation and control systems. Technical Report.

[2] IEC 62443-3-3: Industrial communication networks – network and system security – part 3-3: System security requirements and security levels.

[3] ISO 15408, information technology - security techniques - evaluation criteria for IT security (common criteria).

[4] Abdelkader Magdy Shaaban, Christoph Schmittner, A. B. The design of a divide-and-conquer security framework for autonomous vehicles.

[5] AIT. Threatget - threat analysis and risk management. https://www.threatget.com, 2019. Acessed: 2019-10-20.

[6] Chakraborty, S., Al Faruque, M. A., Chang, W., Goswami, D., Wolf, M., and Zhu, Q. Automotive cyber–physical systems: A tutorial introduction. *IEEE Design & Test 33*, 4 (2016), 92–108.

[7] Ekclhart, A., Fenz, S., Goluch, G., and Weippl, E. Ontological mapping of common criteria's security assurance requirements. In *IFIP International Information Security Conference* (2007), Springer, pp. 85–95.

[8] Ekelhart, A., Fenz, S., Klemen, M., and Weippl, E. Security ontologies: Improving quantitative risk analysis. In *2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07)* (2007), IEEE, pp. 156a–156a.

[9] El Sadany, M., Schmittner, C., and Kastner, W. Assuring compliance with protection profiles with threatget. In *International Conference on Computer Safety, Reliability, and Security* (2019), Springer, pp. 62–73.

[10] Glas, B., Gramm, J., and Vembar, P. Towards an information security framework for the automotive domain. *Automotive-Safety & Security 2014* (2015).

[11] Hernan, S., Lambert, S., Ostwald, T., and Shostack, A. Threat modeling-uncover security design flaws using the stride approach. *MSDN Magazine-Louisville* (2006), 68–75.

[12] KASTEBO, M., and NORDH, V. Model-based security testing in automotive industry. Master's thesis, Department of Computer Science and Engineering - UNIVERSITY OF GOTHENBURG, Gothenburg, Sweden, 2017.

[13] Lévy-Bencheton, C., Marinos, L., Mattioli, R., King, T., Dietzel, C., and Stumpf, J. Threat landscape and good practice guide for internet infrastructure. *EU Agency for Network and Information Security (ENISA)* (2015).

[14] Ma, Z., and Schmittner, C. Threat modeling for automotive security analysis.

[15] Macher, G., Armengaud, E., Brenner, E., and Kreiner, C. Threat and risk assessment methodologies in the automotive domain. *Procedia computer science 83* (2016), 1288–1294.

[16] McAfee. Automotive security best practices. Tech. rep., McAfee, June 2016. Recommendations for security and privacy in the era of the next-generation car.

[17] Member, W. Swrl: A semantic web rule language. https://www.w3.org/Submission/SWRL/, 2004. Accessed: 2019-10-18.

[18] Miller, C., and Valasek, C. Adventures in automotive networks and control units. *Def Con 21* (2013), 260–264.

[19] Miller, C., and Valasek, C. A survey of remote automotive attack surfaces. *black hat USA 2014* (2014), 94.

[20] Mozzaquatro, B. A., Jardim-Goncalves, R., and Agostinho, C. Towards a reference ontology for security in the internet of things. In *Measurements & Networking (M&N), 2015 IEEE International Workshop on* (2015), IEEE, pp. 1–6.

[21] MUTSCHLER, A. S. Data storage issues grow for cars. https://semiengineering.com/data-issues-grow-for-cars/. Accessed: 19-10-2019.

[22] NASA. Your device has more computing power. https://www.nasa.gov/mission_pages/voyager/multimedia/vgrmemory.html. Accessed: 18.10.2019.

[23] NHTSA. Vehicle cybersecurity. https://www.nhtsa.gov/technology-innovation/vehicle-cybersecurity. Accessed: 17.10.2019.

[24] O'Connor, M. J., and Das, A. K. Sqwrl: A query language for owl. In *OWLED* (2009), vol. 529.

[25] Ramesh, R., Prabu, M., Magibalan, S., and Senthilkumar, P. Hazard identification and risk assessment in automotive industry. *International Journal of ChemTech Research 10*, 4 (2017), 352–358.

[26] Recommendation, W. Sparql query language for rdf. https://www.w3.org/TR/rdf-sparql-query/. Accessed: 19.10.2019.

[27] Schikuta, E., Magdy, A., Haq, I. U., Mohamed, A. B., Pittl, B., and Mach, W. Searching the sky for neural networks. In *International Work-Conference on Artificial Neural Networks* (2017), Springer, pp. 167–178.

[28] Schikuta, E., Magdy, A., and Mohamed, A. B. A framework for ontology based management of neural network as a service. In *International Conference on Neural Information Processing* (2016), Springer, pp. 236–243.

[29] Schmittner, C., Gruber, T., Puschner, P., and Schoitsch, E. Security application of failure mode and effect analysis (fmea). In *International Conference on Computer Safety, Reliability, and Security* (2014), Springer, pp. 310–325.

[30] Schmittner, C., Latzenhofer, M., Abdelkader Magdy, S., and Hofer, M. A proposal for a comprehensive automotive cybersecurity reference architecture. In *VEHICULAR 2018, The Seventh International Conference on Advances in Vehicular Systems, Technologies and Applications* (2018).

[31] Schoitsch, E., Schmittner, C., Ma, Z., and Gruber, T. The need for safety and cyber-security co-engineering and standardization for highly automated automotive vehicles. In *Advanced Microsystems for Automotive Applications 2015*. Springer, 2016, pp. 251–261.

[32] Sommer, F., Dürrwang, J., and Kriesten, R. Survey and classification of automotive security attacks. *Information 10*, 4 (2019), 148.

[33] Souag, A., Salinesi, C., Mazo, R., and Comyn-Wattiau, I. A security ontology for security requirements elicitation. In *International symposium on engineering secure software and systems* (2015), Springer, pp. 157–177.

[34] Strobl, S., Hofbauer, D., Schmittner, C., Maksuti, S., Tauber, M., and Delsing, J. Connected cars—threats, vulnerabilities and their impact. In *2018 IEEE Industrial Cyber-Physical Systems (ICPS)* (2018), IEEE, pp. 375–380.