Preliminary Version for the SAE International Journal of Transportation Cybersecurity and Privacy

A Comprehensive Attack and Defense Model for the Automotive Domain

Thomas Hutzelmann, Sebastian Banescu, and Alexander Pretschner
Technical University of Munich, Germany
Email: firstname.lastname@tum.de

Abstract

In the automotive domain, the overall complexity of technical components has increased enormously. Formerly isolated, purely mechanical cars are now a multitude of cyber-physical systems that are continuously interacting with other IT systems, e.g. with the smartphone of their driver or the back-end servers of the car manufacturer. This has huge security implications as demonstrated by several recent research papers that document attacks endangering the safety of the car. However, there is, to the best of our knowledge, no holistic overview or structured description of the complex automotive domain. Without such a big picture, distinct security research remains isolated and is lacking interconnections between the different subsystems. Hence, it is difficult to draw conclusions about the overall security of a car or to identify aspects that have not been sufficiently covered by security analyses. In this work, we propose a comprehensive model covering all relevant aspects of the automotive environment and link it with selected attack scenarios and defense strategies already discussed in academic literature. This showcases the capabilities of our model to build new attack chains, to compare alternative defense strategies, to structure existing work and to identify possibilities for future research.

1. Introduction

About 40 years ago, cars contained just a few basic electronic control units (ECUs) that were mainly connected with actuators and sensors but with little communication between each other. Over time, not only did the amount of software grow exponentially, but also new functionality promoted the concept of an "intelligent vehicle" [1]. Nowadays, we consider a "connected car as a vehicle capable of accessing the Internet [...] of interacting with other smart devices on the road or in mechanical shops [...] [and] with other vehicles." [2]

These new levels of interaction and variety of technologies give rise to new security problems and concerns. Security flaws that are not dangerous in isolation can be combined to enable multistep-attacks and thereby raise serious security concerns which imply safety flaws for passengers. In this work, we present a model that sheds light on these attack chains and their implications.

We address the *problem* that today's automotive environment has grown into a complex interconnection of diverse systems (e.g. ECUs, infotainment system, remote services, garages). To the best of our knowledge, there is no holistic overview nor any structured description of this environment, which includes grave attacks and the defenses against them.

We propose a comprehensive model as a *solution* to this problem and to enable the description and discussion of multistep-attacks and the underlying weaknesses. It allows us to represent and visualize attacks and defenses presented in the academic literature in a formal, uniform way.

The *contribution* offered by this paper consists of a comprehensive model deduced from the academic literature and discussions with automotive professionals. Furthermore, a mapping of selected research about automotive attacks and defenses shows the applicability and merit of our model, which is designed to: (1) organize and structure academic literature and related work, (2) highlight gaps in existing research, (3) compare alternative defense strategies and (4) combine described attacks into new attack scenarios.

The paper is structured in the following way: Section 2 presents all the elements that constitute our suggestion for a comprehensive model. Section 3 places our model in the context of related surveys. Section 4 spotlights the merits and advantages of using our model Section 5 concludes with an outlook on future work.

2. A Comprehensive Model

To manage the growing numbers of technologies and research relevant to the automotive domain, we consider a single, integrated model linking all core aspects to be a valuable navigation tool. However, due to the high complexity of the domain, a comprehensive model needs a substructure for convenient handling and easier understanding. Therefore, we propose three sub-models covering: (1) environmental aspects, (2) the steps of an attack and (3) defense mechanisms. These are linked together to enable a comprehensive view of the security level of today's cars. We start their discussion with a detailed look at each of the submodels and integrate them afterwards. The big picture showing the connections between the submodels is given in Figure 5 and can be used for better orientation in this section.

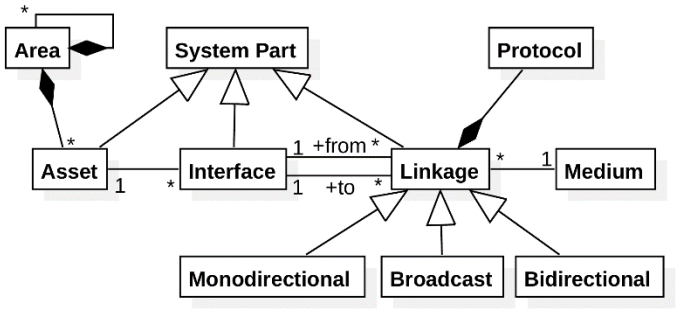

Figure 1: High level model of the Environment

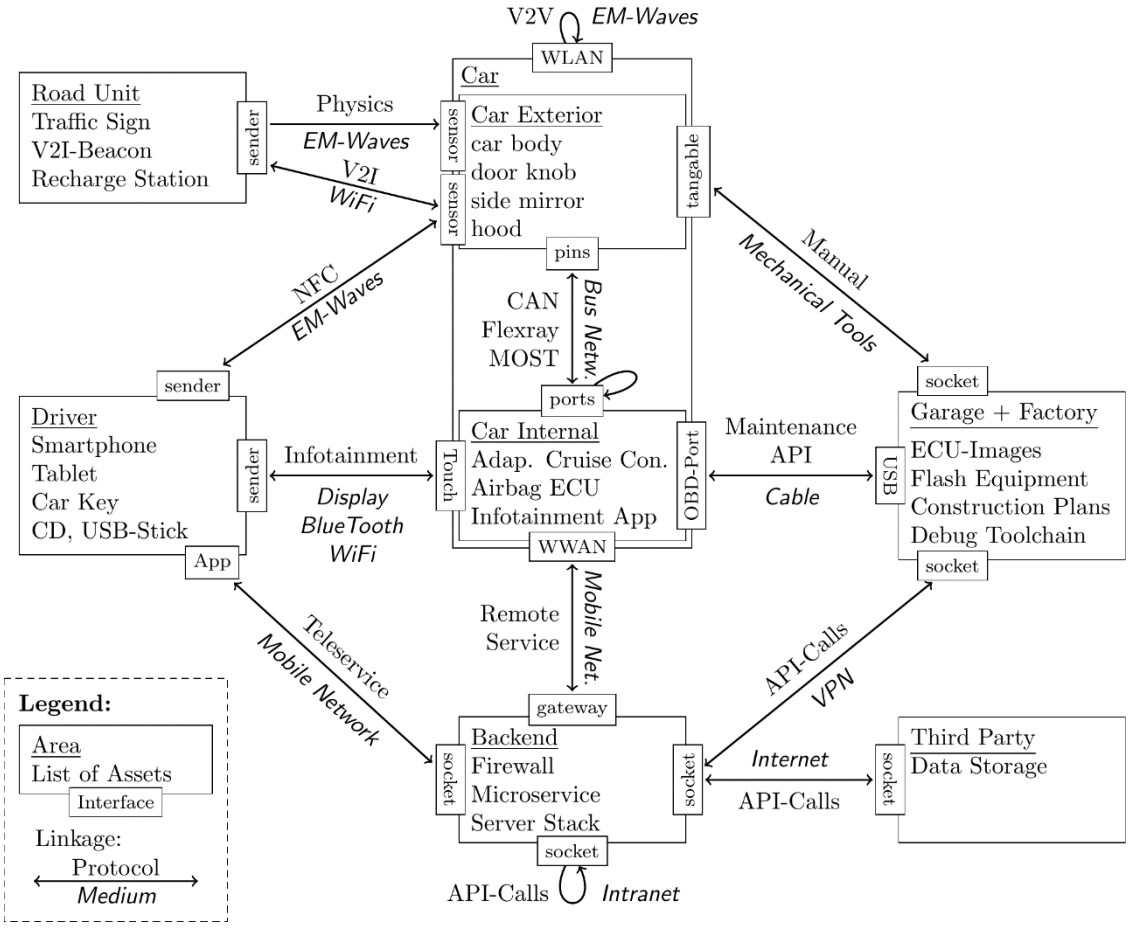

Figure 2: Concretization of the Environment Model

We consider this model and the mapping as a valuable contribution to both academia and industry. Researchers can use this model to systematically classify and group existing academic literature. This is especially helpful for related work and for identifying research gaps. Related research is classified similarly and work uncovering new attack-vectors in the system inherently requires new defense mechanisms that can be linked with them. For industry experts, this model can be helpful to explore security problems and solutions. Using the interconnections and the attributes within the model, they can compare attacks with their prerequisite and potential impact on the defenses with their effectiveness and the efforts (i.e. costs) involved in implementation. We believe that this is a helpful guide for decision-makers when it comes to incorporating a higher level of security into the automotive domain, as presented by researchers.

In the following sections, we walk through the three submodels. For making the explanations more tangible, we are elaborating a hypothetical example of an attack. In this scenario, an adversary uses a malicious smartphone app to gain remote control over a car.

2.1 Environment Model

All security analyses can only be carried out with respect to the concrete, underlying system in the specific context of a given environment. Hence, the foundation for our model is formed by a model for the automotive environment containing the relevant system parts. We are aware of the continued development and changes in the automotive domain. So a single static model can ever only be a snapshot of the environment, which sooner or later will become outdated and would require adjustments in the model.

Therefore, we provide two models: A higher level model defining the basic structure of the environment together with its interrelations and a specific representation of this model that depicts the current state-of-the-art in the automotive domain. Future security analyses can adapt and extend the specific instances, although the higher level model remains constant.

2.1.1 High Level Environment Model

A UML class diagram for the complete model is illustrated in Figure 1. Today's automotive environment can be split into distinct *areas*, e.g. the car itself and its surroundings on the road. Each of these areas is composed of different *assets* (e.g. ECUs or a micro-service) and may have nested areas supporting a deeper, more detailed analysis. These assets may have a varying number of *interfaces* that connect them with other assets from the same area or from some other area. Such a connection is called *linkage* and can be *monodirectional*, *bidirectional* or a *broadcast* communication. Each of connections requires a *protocol* for communication and a *medium* for transporting the communication. As assets, interfaces, and linkages all potentially exhibit attack surfaces, they are generalized as *system parts* and referenced as such in the attack model.

2.1.2 Concretization of the Environment Model

While the abstract model ensures an immutable foundation for the environment description, it needs to be specialized on the interrelations in the automotive domain for any concrete security analyses to be possible. To tailor it to the domain, we instantiate this model with the relevant entities from the automotive ecosystem.

We focus mainly on the diversity of the areas and added enough details to map ongoing research to the model, but we skip the technical embodiment of all the assets, in order to limit the initial complexity of the model. If needed, domain experts can add more levels of detail for their special use cases. As this enrichment happens based on the higher level model, the comprehensive view retains its connections and usability.

Our initial concretization of the model is illustrated in Figure 2. The most important area in the automotive environment is the car. Due to its complexity, we split this area into two subareas: *car internals* and the *car exterior*. The *car exterior* is a collection of everything that is tangible when you look at a car, e.g. the car body, a doorknob or the hood. Its counterpart consists of the *car internals*, everything that is hidden out of sight in the interior of the car, e.g. the body control module, the motor and apps for the infotainment system. The entity interacting the most with the car is the *driver*. The driver can use all the offerings of the infotainment system (e.g. the displays) and connect various assets (e.g. a smartphone, Bluetooth speaker, for more details see Oka et al. [3]) to the car. The driver herself is not an asset, as our model works on information assets only. However, she is indirectly represented by the assets she is using and their design. Another area consists of everything on the *road*, for example, the hardware required for vehicle to road infrastructure (V2R) communication (for a more detailed description see Lu et al. [4]). Finally, there are several areas related to the manufacturer of the car. First, everything that is needed to build and maintain the car is grouped in the *garage & factory* area. This includes the construction plans, the images for the ECUs and the whole debugging tool-chain. Furthermore, modern cars use the mobile telelphone network to establish a steady connection to a *backend,* operated by the respective manufacturers. This backend is used to provide additional services to the driver. The assets of the *backend*-area are the software components required to provide these services, e.g. the server operating system or the micro-services. Most likely, not all of these services can be provided by the manufacturer alone. Hence, the backend is connected to the *third party* area that can provide additional services, for example, updated maps or traffic information.

In our hypothetical example, we need to add different, concrete assets to the model. The attack starts with a malicious smartphone app, so the driver needs to use it on her smartphone and have a link to the infotainment app. For gaining remote control, the access of the attacker needs to be extended to the head unit, and via the CAN bus to the ECUs that are actually controlling the car.

2.2 Attack Model

After describing the different systems, our next sub-model focuses on the attacks on these systems. This model is illustrated in Figure 3. The core node of the attack-model is the *attacker* performing the attack. Initially, each attacker has several *capabilities* that allows them to start the attack. This can be some level of *access* to the car or the ownership of and knowledge about *tools*. Other capabilities gained or required by exploits are labeled as *aptitudes*. These initial capabilities enable the attacker to execute an *exploit* on some system part. If this exploit is successful, the attacker may gain additional capabilities from the infected component. These new capabilities can enable the attacker to execute other exploits on some other or the same system part. This procedure continues until the attackers reach an exploit that allows them to achieve their final *objective*. A successful attack is not just a single exploit, but a consecutive chain of exploits depending on each other, building up more capabilities for the attacker until they achieve their objective.

Our model enriches this core skeleton with additional information and classification that gives a better understanding of the attack. Each attacker has a *motivation*. The most harmless attackers just want to tinker with the car. The reasons for this might be personal satisfaction or to enhance their reputation in an arbitrary community. Research falls into this category. Another group of attackers seek financial profit, e.g. by stealing components or selling "unofficial" software updates to car owners. Finally, there are attackers who endeavor to cause maximum damage, financially or to other people.

In addition to the motive behind it, each attack is also aimed at a concrete *objective*. These objectives can take various forms and may be represented as sub-classes of the objective. At an initial level, they can be separated into *in-system* objectives and objectives with *trans system* effects. On the one hand, in-system objectives are directly linked and located to specific system parts, e.g. the theft of a component, car tuning or violation of the safety properties of system components. On the other hand, trans-system objectives are related to transcending units, which cannot be located directly in any one part of the system, e.g. the exfiltration of data, which is transferred through the system, the intellectual property of specific architectures or solutions or, in general, the reputation of the manufacturer. Some motives have direct, implicit objectives. For example, vandals will mainly focus on sabotaging the car, e.g. they will destroy the car completely or just disable safety-critical components. The tinkerer aims to modify their car. This includes classical car tuning but extends also to unlocking "premium features". We expect enabling these features without paying to become increasingly significant in the future [5]. However, there is no distinct mapping between the

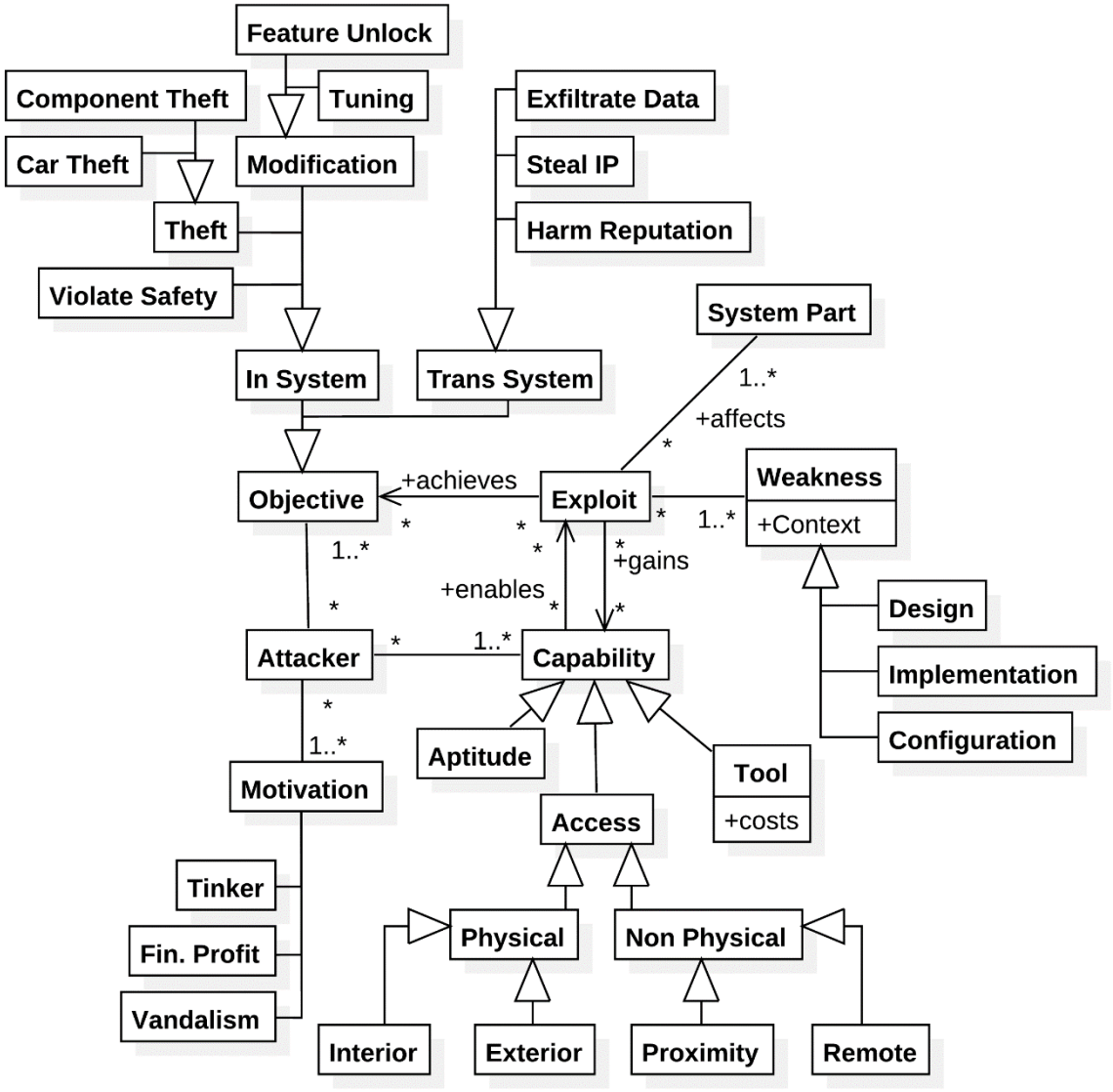

Figure 3: Attack Model

sub-types of motives and objectives: For example, an attacker can use an exploit endangering safety to blackmail the manufacturer, and thus for his own financial gain. The objectives and motives of attackers can be combined in various forms and more examples can be found in academic literature [6]. For *capabilities*, it is most important to separate different levels of access to the car. The OBD-II port exposes a big attack surface, but it requires physical access to the interior of the car that is only available to a small group of people, e.g. the car owner, passengers or technicians in the repair garage. More likely are exploits on web-services of the backend, as they are remotely available over the Internet. Physical access to the exterior of the car also offers lots of attack vectors, e.g. breaking into the car, but entails the risk of being discovered and arrested during the attack. Exploits on sensors require physical proximity to the car, e.g. being on the same street as the attacker.

As exploits are the pivotal element of our attack model, we enhance them by adding different classes of exploit. First of all, each exploit affects at least one system part. The system part class is identical to the same-named class in the previous environment model. In this way, an exploit can affect assets, interfaces, and linkages in all areas of the system. In our model, each exploit is also based on a weakness in a system part that it exploits. This weakness can be in the design, the implementation or the configuration of the system part.

For our fictional attack scenario, we have to be more concrete about the actual steps of the attacks. While remote controlling the car, the attacker might aim for violating the safety, for example, by actively crashing the car; or she can harm the reputation, for example, by selectively degrading features like the adaptive cruise control. Both are motivated by doing vandalism. Next, we need to define the exploit-capability chain: The attack is starting with the remote access of the attacker to the smartphone of the user. This capability could be gained via a malicious app in the app store, which is not part of the model presented in Figure 2. In this model, as an initial capability, we take the ability of the attacker to execute code on the smartphone. From there, the attacker can also send crafted messages to the Bluetooth interface of the car, while the driver is inside. There might be weaknesses in the implementation of the communication stack, for example, a buffer overflow that enables the attacker to execute code on the head unit. Thereby, she can access the CAN bus and can send messages to other ECUs inside the car. From that point, there are many alternatives for harming safety, see section 4 for details. One example is to spam packages and with this deny all legitimate communication between the sensors and ECUs connected to the bus.

2.3 Defense Model

Finally, the third model is dedicated to possible defense mechanisms. It is illustrated in Figure 4. Such mechanisms can be classified by multiple relevant properties. First of all, the model considers the *implementation* of the defense, as this can be done in *software* or at *hardware level*. Each defense can have multiple, different *effects*: Ideally, it *solves* the underlying problem, so that the corresponding attacks become impossible. For example, bug fixes in the implementation of software should have this effect. Other defense mechanisms cannot prevent attacks completely, but at least they can *harden* the system. All cryptographically secure encryption algorithms have such effects, as in theory decryption is always possible by brute forcing the key; but practically, the required computation power to perform the attack is not affordable. As subclasses of the hardening group, we reuse the three aspects of the CIA-triad, *confidentiality*, *integrity* and *availability*. Please note, that CIA is intentionally not represented in the attack model. The attack model focuses on describing an attacker that is exploiting weaknesses to gain more capabilities until she reaches her final objective. These exploits are violating the CIA attributes, and often several of them are affected simultaneously, for example when an ECU is infected via exploiting a buffer overflow. In this example, the attacker gains access to CAN bus, where she can read all messages (confidentiality), manipulate sent signals (integrity) and flood the bus to make it unusable (availability). However, these attacks in most cases are only based on message manipulation, so hardening by using cryptographic signatures is a valid defense. With these links, the relevant CIA attributes of the attack are implicitly described by the linked defenses. Finally, the minor effect any defense can have is to raise *awareness* of the attack. This means that even though the defense does not prevent or stop the attack, it does give an indication that the attack is ongoing or has happened, so that its causes can be mitigated. One important factor that can be used to compare different defense mechanisms dealing with the same attack is the cost of their implementation. Cost is integrated into our model via the indirection of *overhead* of a defense. These overheads can manifest in *memory*, *performance*, the *development* effort to implement the defense. It is also possible to have a general overhead without the refinement of one of these subclasses. For example, physical unclonable functions are a defense mechanism that can only be implemented in hardware [7]. Thus, they do not directly increase the overhead in memory or performance but do cause a cost to the specialized hardware. Lastly, all defenses follow a *strategy*. On the highest level, a defense strategy can be classified as *proactive* – measures taken against attacks before they happen – and *reactive* – measures taken against attacks while they are happening or after they have happened.

Proactive measures can be further refined in strategies aimed at *non-functional* or for *adaptation* of the system. The first category, hardening, is based on adding additional layers to the software, like applying *cryptography* before the communication or storage of data or *integrity protection* that adds tamper-protection mechanisms to existing software or data. The second category, adoption, incorporates the changes directly into the system. The most basic case is just a*bug fix* that repairs a flaw in the implementation or configuration. Furthermore, the system can be adapted to *limit access* to a sensitive component, e.g. by restructuring the network or restricting it with a firewall. All reactive measures provoke a *reaction* whenever they detect an attack. These reactions can be automatic, e.g. a script adjusting a firewall, or manual e.g. an alarm alerting a human security agent who will conduct further investigations of the incident. The minimal reactive measure is just *monitoring* the system. These defenses, e.g. recording performance indicators or storing some log files for later manual analysis, are based on some information about the system but do not apply continued scans automatically.

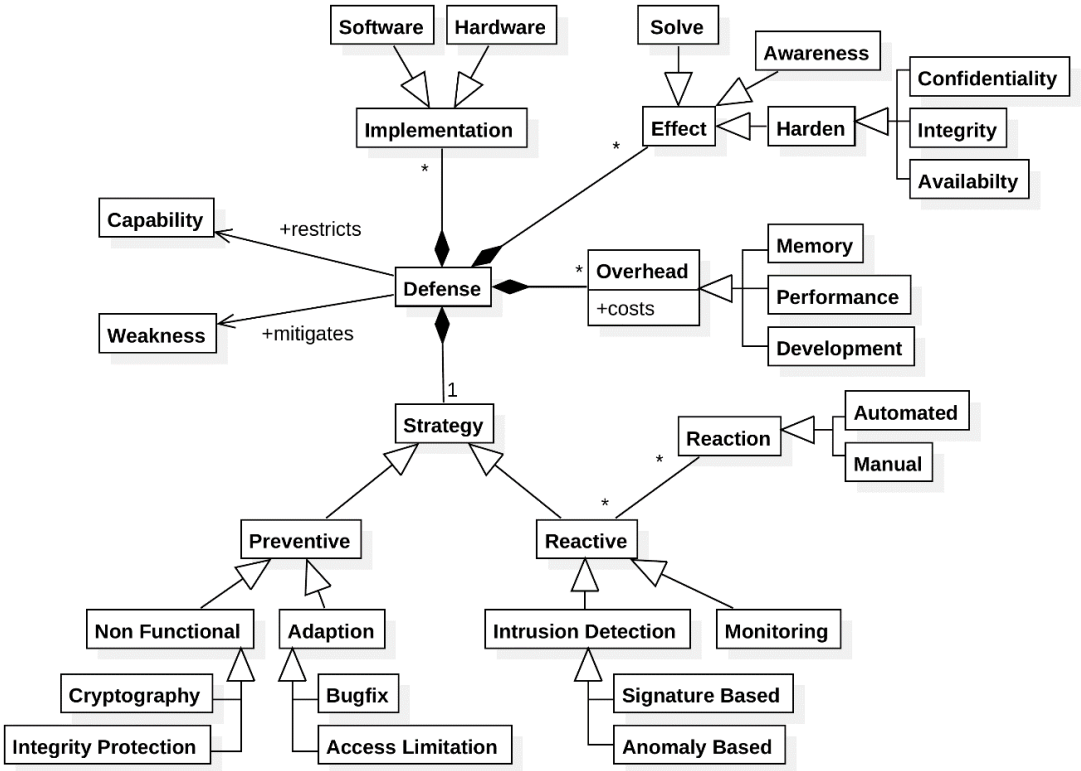

Figure 4: Defense Model

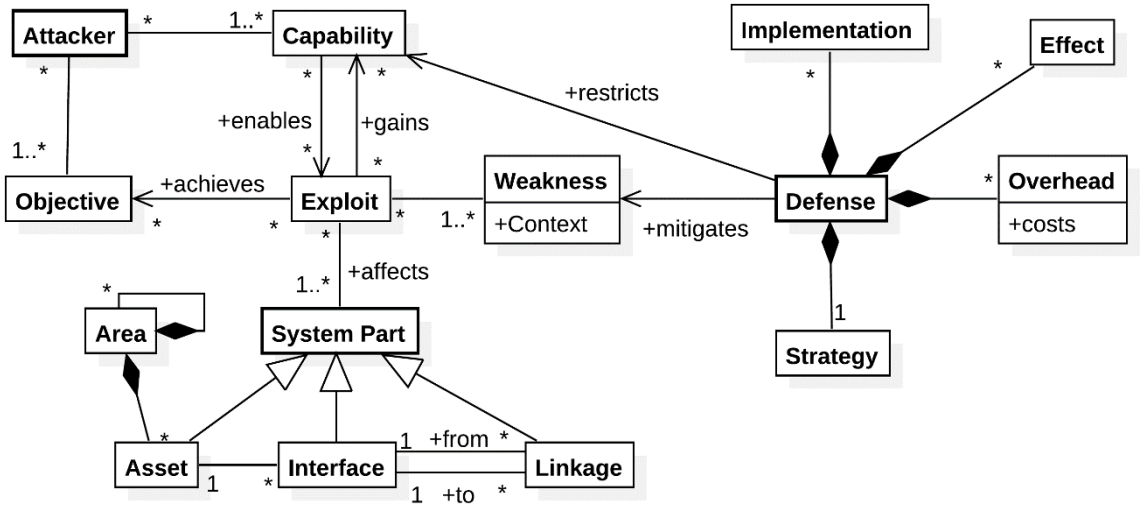

Figure 5: Interconnections between the Models

More advanced reactive measures are classified under *intrusion detection* (e.g. as applied by Hoppe et al. [8]). Depending on its internal mechanics these defenses can be refined in two subclasses. On the one hand, *signature based* defenses have information about previous attacks and have knowledge about concrete indicators of an ongoing attack. This knowledge is used to define signatures of the attack (e.g. rules, patterns) that are then used to scan ongoing actions. On the other hand, *anomaly based* defenses do not have information about attacks and extract or learn rules from the regular behavior of the system and consider deviations as abnormal. These two subclasses are common in academic literature: for example, see Axelsson [9] and Kabiri et al. [10]. The nodes *capability* and *weakness* constitute the interconnection to the attack model and will be discussed in more detail in the next subsection.

There is intentionally nothing specifically automotive in our defense model, as we do not see a requirement for it. Software engineering is all about different trade-offs and the defense model describes these. The decision as to which trade-offs are acceptable is always domain-specific. However, these decisions are not represented in the model and thus, the defense model does not specialize in the automotive domain.

For our fictional example of an attack via a malicious smartphone app, various possible defenses could be modeled. The most basic example is fixing the buffer-overflow weaknesses in the Bluetooth implementation that solves the attack vector with no noticeable overhead. Furthermore, the architecture of the internal bus communication can be adapted to clearly separate the head unit from any other safety-critical communication. However, this only hardens the integrity of the system, because as long as the buses are connected, an attacker just needs to perform more exploits to reach the same capabilities again. Furthermore, that particular architectural change has to be realized in hardware and therefore, it is very expensive. Namely, it might be more feasible to add a reactive measure. For example, an additional software component in the central gateway that detects anomalous traffic, as a reaction—isolates the affected network or component; reports this to the driver to raise the awareness about the ongoing attack and sends the car to a repair shop. All these are possible defenses, that can be compared using the properties in our model and the manufacturer, finally, could decide which should be implemented.

2.4 Interconnection of the Models

After discussing all three sub-models in detail, this subsection now focuses on the interconnection of the models. The connection between the models is depicted in Figure 5. The environment builds the foundation of our model. With the proposed higher level model, almost every part of the environment is a system-part. No matter in what area it is used or how the assets are connected, every part of the environment can be exploited for an attack. If there exists an attack, the abused system-part has a direct linkage to the attack model. This allows us to query the combined model as to which system part is attacked more frequently or which attacks need to be considered when a specific part of the environment is designed or implemented. Summarizing the attack model, a successful attack in our model is a chain of capabilities and exploits that finally achieves an object. Hence, any successful defense must break this chain. This can be done in two ways: Either the defense restricts the capabilities of the attackers, so that they are no longer able to perform an exploit; or else it mitigates the underlying weakness, such that the attackers can no longer use it to gain more capabilities. Intentionally, our model has no direct connection between the system and the defense. This is in order to always require a concrete attack for such linkage. Accordingly, there are two ways to model defenses: Either they are part of the system or else they function as an asset of the environment. Defenses designed to thwart a specific attack should be modeled as an explicit defense against this attack.

Otherwise, the “defense” has to be modeled as an asset inside the environment that inhibits an attack surface, just like any other part of the environment. In other words, a defense should at least have one specific attack scenario that it protects against to qualify as a defense.

With splitting the model into three submodels and with the generalization to a generic representation, we aimed for making the complexity in the automotive domain manageable. Our model is applicable to older cars without ECUs up to completely autonomous cars. Modern architectures are composed of a multitude more aspects, but just modelling a single attack on a single component is sufficient to initialize and use our model. Further links, attacks, and defenses can be added when required or needed.

3. Related Work

Our initial models were inspired by the thoughts about the automotive threat model of Checkoway et al. [11]. For their threat assessment, they focus on the capabilities of an attacker that enables them to execute an exploit, e.g. the ability to write an IDA Pro module for modifying the firmware of an ECU. Furthermore, they show several attacks using chains of multiple exploits that are also reflected in our attack model. However, they choose a different approach to classifying the level of access and the capabilities of the attacker. They structure the access level into "indirect physical", "short-range wireless" and "long-range wireless" areas. In contrast to our model, they are not considering remote attacks like attacks on the OEM servers that do not require any proximity to the car. Additionally, their structure is based on the physical properties of the transport medium for the attacks. Hence, such a model cannot be applied to classical attacks like cutting cables or lock-picking. We think classification of the access level by the distance of the initial interaction with the car is more suitable, since this is independent of the medium used by the attacker and thus more enduring for future developments. Finally, they distinguish between "technical", the knowledge and skills of the attacker, and "operational" capabilities, the level of access an attacker can achieve. Our separation between tool and access sounds similar at the first sight; but we consider separating exploits by the required knowledge of the attacker as a variant of security by obscurity. Hence, it is barely helpful for classification purposes. As an alternative, we suggest using the notion of tools that are required to execute the corresponding exploit, as a tool is always related to the cost involved in building or buying it. This cost makes some of the exploits unaffordable for some attackers. For example, it is not feasible for a car tuner to spend a lot of money on expensive equipment for illegally unlocking a premium feature that is comparably cheap to buy directly. Unfortunately, none of authors' thoughts are formalized or generalized and they do not provide a structure for their recommended defense mechanisms.

Another survey that has helped to refine our model was done by Studnia et al. [12]. In their study, they are naming and grouping a lot of elements that need to be considered in security analyses. They start with a detailed overview of the automotive networks that conforms to our environment model. Then, they classify the goals of the attacker. These are represented in more detail in the subclasses of the objective and the motive of the attacker in our attack model. Furthermore, the survey lists several attacks grouped by their level of access and covering different elements in the environment (e.g. the OBD-II port, the TPMS or the Web browser in the infotainment). Finally, the authors list several defense mechanisms, like anomaly detection and cryptography, that are also represented in the strategy subclasses contained in the defense model. Their survey gives an overview of the interconnections between environment, attacks and defenses. Our model combines the different views on security given by them into one interconnected model that provides enough structure for them to be able to merge their work with similar and future research.

Finally, our model is a more formalized view of the landscapes drawn by the survey of Parkinson at al. [13], which focuses on research gaps and future challenges. Their initial outline is similar to our environment model, as they also consider the car, the road environment. and the connections between them all as distinct elements. However, our model also considers the manufacturer and the backend for connected services as parts of the environment that may contain new challenges for research. Subsequently, they outline a total of 14 research gaps that can (with one exception—privacy concerns—these can only be considered from an attacker's perspective as data leakage) all be located in our model since, while designing our model, we intended to cover all of their findings. Three gaps are related to attacks on sensors, the effects of compromised sensors and protections against these attacks. Most central for us is the lack of comprehensive research on ECU vulnerabilities, their combinations, potential implications and mitigation techniques. The interconnection of exploits and capabilities of the attackers as well as the interconnectivity of attacks and defenses are designed to enable a mapping that highlights this gap. Two other gaps, namely research into a secure update mechanism and also secure forensic storage, are considered as assets at the backend of the manufacturer and inside the car, because they are not only defense mechanisms but their security needs to be discussed in the context of the entire system and known attacks. The third party information in the infotainment system is represented in our model as a trace from the third party area through the backend into the infotainment system as we consider this level of detail as highly relevant to discuss these attacks. Sensors are part of our environment model, hence it can be used to discuss all of the three gaps. Furthermore, three other gaps involve considering the reactions to and the detectability of attacks. Our defense model differentiates multiple defense strategies and the reaction should directly be linked with each of these strategies. Finally, the three remaining gaps point to a lack of industry adaption and reaction to automated attacks by non-professional attackers. Moreover, industry should put more effort into fixing inherent security vulnerabilities in the components. Our model includes the notion of costs for attack tools and defense mechanisms to quantify these aspects. Hopefully this will make this problem more tangible for the management level and will increase the efforts in this research direction.

4. Application of the Model

After the discussion of the delta from our model to previous work, below we will elaborate the merits of using our threefold model. Mainly, we see these benefits in four different aspects. Each of them will be discussed in one of the following subsections.

4.1 Structuring existing academic literature

First of all, our model provides a structured view of the mapped academic literature. Due to time and space limitations, it is not feasible to give an exhaustive mapping here. Hence, we have selected academic literature from attacks and defenses to cover all aspects of our model at least once. When multiple papers were available, we choose the most cited source. By this process, we demonstrate the applicability of our model to the current research landscape. Our mapping can be found in Table 1.

In this table, each column represents one class from our model. Namely from left to right: the environment model, the attacker model and the defense model – each separated by double lines. To prevent proliferation of the number of columns, we skipped all inherited classes and just kept the parent class and give the concrete subclass in the table fields. Each row represents a pair of attack and correlated defense strategies documented in one piece of academic literature. The first column cites the reference for the illustrated instance. If one reference explains multiple attacks and defenses, these are listed one below the other and only the first row cites the reference.

In order to keep this table compact and on a single page we use abbreviations and some symbols inside the table. Words ending with a dot are short versions of class names in our model, chosen long

enough to match uniquely to one subclass. A field with "n/a" indicates that the corresponding information was not given by the authors in this reference, but would be required for a complete instantiation of our model. A field with "–" indicates, that this field is not needed for this specific instance. This is only used for the reaction only required by a reactive defense strategy.

We want to point out that our collection ranges from survey papers over experimental analyses to blog posts on the Internet. Although all of them give different granularity of information and level of details, all of them are still mapable to our structure. This underlines the capabilities of our model.

Such a structured view is helpful for automotive professionals as well as academic research. On the one hand, domain experts are mainly responsible for one specific part of the car, e.g. one functionality inside an ECU or one remote service inside the backend. The explicit environment model allows them to filter for their component and only focus on information available for their field of interest. On the other hand, researchers need a broad overview of all existing research to identify gaps and link their work to closely related previous publications. Our model directly links attacks with defenses and vice versa. Thereby research on attacks can show that it circumvents existing defense strategies and new defense mechanisms can rely on documented attacks to prove their effectiveness. Hence, we consider our model to be a valuable aid to the collection of related work.

4.2 Comparison of alternative defenses

In our model, a description of an attack is split into exploits and different levels of capabilities i.e. all intermediate steps of the attack have to be explicit. In this way, not only are the attacks illustrated with a high level of detail, but this also enables a very detailed discussion of different approaches for a suitable defense. Longer attack chains, as for example described by Spaar [17], hold more potential for defenses. The single strategy discussed in their work is just preventing the final replay step of the attack.

Even though their work focuses on the description of the attack, this attack can however be impeded by addressing each segment of the attack chain.

If the initial soldering of the debug pins on the chips inside the ECU that have been intentionally removed is impeded, no meaningful debugging of the firmware is possible and the vulnerabilities exploited will be harder to discover. The weak DES encryption used by the chip could be replaced by its successor 3DES or by stronger AES encryption; which would harden the confidentiality of the sniffed messages, but would also involve higher computational costs. When switching the cryptography, it is also possible to use modern, white-box cryptography [26]. This would hide the key inside the implementation, so that it can no longer be discovered easily by code inspection. At the simplest level, sensitive information like the VIN of the car should not be divulged inadvertently, e.g. by verbose error messages. This small change in implementation would require an additional step in the attack chain in order to gather this information. Additionally, the attackers would have to use their own mobile station for sending fake messages to the car.

By fingerprinting mobile stations that have been used previously, and also by collecting a pool of trusted stations, the manufacturer could adopt a reactive strategy of restricting the use of "unknown" stations, thus preventing a car unlock command via this channel, for example. Finally, the car owner is not included at any point in the attack chain. Therefore, an automatic information notice on a third channel, stating that the car has been unlocked by the remote service, would provide notification of the attack.

These defenses would be capable of mitigating the attack described, since each of them breaks the chain of exploits. However, each defense involves a different cost and has a different impact – not only on an individual attack but also on the overall level of security. Furthermore, in our opinion, security mechanisms always need to consider the architecture of the entire system: The car's internals and the road units are just as important as the manufacturer's backend and repair garages. All of these areas are needed when it comes to designing a holistic, fully functional security architecture. Hence, we consider this discussion to be a valuable application of our model.

4.3 New combinations of existing attacks

The core element of our attack model is the circular relationship between capabilities and exploits. A successful attack starts with the initial capabilities of each attacker, followed by the exploits these enable and continues similarly with new capabilities gained through these exploits. Each exploit and the capabilities it requires are just one segment of these attack chains and can be reused for other attacks, although they have been described by different authors. One example of such interaction is the work done by Computest [19] and the work of Koscher et al. [15].

Computest focuses on analyses of the cellular connection, the interface to the modular infotainment platform linking it with the Internet. They have been able to exploit one of the running services, so as to enable remote code execution. Furthermore, they broke the weak encryption keys inside the ECUs and finally were able to send messages over the CAN bus. Computest stopped their research at this point and disclosed its findings to the manufacturer. The manufacturer confirmed their findings but points out that the final "objective of manipulating the steering and brakes was not achieved". In this situation, our model provides more information and links work that shows how an attacker can continue with the ability to send arbitrary CAN messages; namely, the work of Koscher et al. They started their security analysis with physical access to the OBD-II port in the interior of the car and the ability to send messages using it. With exploits on several connected ECUs, they were able to bridge different CAN networks, composite their attacks and finally managed to interfere with the brakes of the car. Both attacks when combined form a new, even more dangerous attack chain. In combination, the attack only requires remote access to the car via the mobile network and achieves the objective of active interference with a car in motion.

This is only one example for such a combination of attacks, but it shows that structuring existing research provides stronger arguments for persuading manufacturers to take additional security steps; especially as, according to the findings of Computest, existing production cars have still not been patched.

Table 1: Structuring selected existing research according to our threefold model.

Ref	Area	System Part	Moti-vation	Objec-tive	Access	Tool	Aptitude	Exploit	Weak-ness	Strategy	Implemen-tation	Effect	Over-head	React.
[11]	Car Internal	CD-Player	n/a	n/a	Interior	IDA Pro	Send CAN	ECU Flash	Config.	n/a	n/a	n/a	n/a	-
	Car Internal	Media Player	n/a	n/a	Interior	IDA Pro	Send CAN	Buffer Overflow	Implemen.	n/a	n/a	n/a	n/a	-
	Garage+ Factory	OBD-PassThru	n/a	n/a	Proxim.	WiFi Scanner	Send CAN	Shell Injection	Implemen., Config.	n/a	n/a	n/a	n/a	-
	Garage+ Factory	Windows Laptop	n/a	n/a	Remote	Virus for Windows	Access to garage Network	Arbitrary	n/a	n/a	n/a	n/a	n/a	-
	Car	Blue Tooth Device	n/a	n/a	Proxim.	IDA Pro	Send CAN	Brute Force, Buffer Overflow	Design, Imple-men.	n/a	n/a	n/a	n/a	-
	Driver Device	Smart-phone	n/a	n/a	Remote	Malicious App	Blue Tooth access, Send CAN	Arbitrary	-	n/a	n/a	n/a	n/a	-
	Car	Telematics Unit	n/a	n/a	Remote	Reverse Eng.	Send CAN	Brute Force	Config.	Bugfix	Software	Awareness	n/a	-
[14]	Backend	Mobile Network	n/a	n/a	Remote	Fake Mobile Station	Send CAN	Reverse Eng.	Config.	Bugfix	Software	Solve	n/a	-
	Car	Blue Tooth Device	n/a	n/a	Proxim.	n/a	Send CAN	Fuzzing	Implemen.	Bugfix	Software	Solve	n/a	-
	Car	OBD-II Port	n/a	n/a	Interior	OBD-Device	Send CAN	Reverse Engineering	Implemen., Configu-ration	Bugfix	Software	Solve	n/a	-
[15]	Car	OBD-II Port	n/a	n/a	Interior	Carshark, IDA Pro, OBD-Device	Send CAN, Reprogram ECU, Control Displays, Influence Safety, drive car	Fuzzing, Jamming, DoS,	Design	n/a	n/a	n/a	n/a	-
[16]	Car	Sensors	Vandalism	Sabotage	Proxim.	Arduino, Signal Generator, Laser	Influence Safety	Jamming, Spoofing, Overload	Design	n/a	n/a	n/a	n/a	-
[17]	Backend	Mobile Network	Financial Profit	Car Theft	Proxim.	Fake Mobile Station	Unlock Car	Message Spoofing	Config.	Bugfix	Software	Solve	n/a	-
	Car	Telematics Unit	Tinker	Data Leakage	Interior	IDA Pro	Read Backend Communication	Brute Force	Implemen.	Bugfix	Software	Harden	n/a	-
[18]	Car	CAN	n/a	n/a	n/a	n/a	Send CAN	n/a	n/a	Anomaly based Intrusion Detec-tion	Software	Awareness	n/a	n/a
[19]	Car	Mobile Network	n/a	n/a	Remote	Port Scanner	Send CAN	n/a	n/a	n/a	n/a	n/a	n/a	-
[20]	Car Internal	Airbar ECU	n/a	n/a	Interior	OBD-Device	Influence Safety	Unauth. Usage, Brute Force	Design	Crypto	Software	Harden	n/a	-
[21]	Car	Multiple	n/a	n/a	n/a	n/a	ECU Flash, Unlock car	n/a	n/a	Anomaly based Intrusion Detec-tion	Software	Awareness	n/a	n/a
[22]	n/a	n/a	n/a	n/a	Remote, Proxim.	n/a	Send CAN, Reprogram ECU, Influence Safety, drive car	n/a	n/a	n/a	n/a	n/a	n/a	-
[23]	Driver Device, Car	OBD-II Port, car key	Financial Profit	Car Theft	Proxim., Inte-rior	OBD-Reader, Keycode Fuzzer	Drive Car	Brute Force	Design	Access Limi-tation, Crypto	Software	Solve, Harden	n/a	-
[24]	Driver Device, Backend	Smartphone App, Rest-API	Vandalism	Data Leakage, Reputa-tion	Remote	Webbrowser	-	Unauth. Usage	Design	Access Limita-tion	Software	Solve	n/a	-
[25]	Car	Wifi	n/a	n/a	Proxim.	Wireshark	Unlock Car	Brute Force	Design	Access Limita-tion	Software	Solve	n/a	-

4.4 Spotting gaps and potential for further research

With more and more research focusing on exploits or the defense of specific parts of the automotive domain, e.g. on inter-ECU communication, some of the components needed to describe attacks are redundant, e.g. the lack of authentication in the CAN bus as a weakness or detrimental to the safety of the passengers who could be the objective of an attack. Koscher et al. [15] find a remarkable solution for tailoring the description of the attacker to their work, as they "intentionally and explicitly skirt the question of a threat model". For research as focused as theirs, skipping this is valid, but for a comprehensive security analysis, the threat model is vital. In our model, we have included the attackers, their motivation and the objective they want to achieve. They are directly linked to our attack chain and so they are reusable. Thus, in cases where the authors do not directly provide this information, our model is a useful add-on, as it spots missing information. Clearly scoped, exhaustive work can also profit from the context given by our model. The work of Yan et al. [16] focuses on attacks on various types of sensors that manipulate their readings. However, they do not take into consideration the different types of attackers and what they can achieve with these methods. Can a speedster use a manipulated sensor to trick an autonomous car in a more aggressive, faster operation? Or can a portable "attack-device" at a crossing cause passing cars to crash without leaving any traces once the device has been removed? The attacker's perspective offers new views on research and makes them more tangible for the car industry. Furthermore, our model requires a variety of details from different aspects of an attack or defense. Unfortunately, most of the academic literature reviewed does not provide sufficient information to fill all classes of our model. As can be seen in Table 1, descriptions of attacks rarely discuss the motive or the objective of the attack. In particular, the granularity of the objective in our model is never considered in research, e.g. we were not able to find any academic literature about car tuning or about the theft of intellectual property. Also, information about the cost of a defense mechanism was not given for any of suggested defenses. The research proposing reactive defense strategies in our selection does not offer any specific reactions other than notification of the user.

Finally, we noticed a remarkable development in research on sophisticated defense strategies, e.g. the work of Kang et al. [18]. They develop an advanced defense mechanism based on anomaly detection, but their work is only linked with "a general attack scenario" and does not give any information about the specific context of the attack used in their analysis. We consider attacks and defenses as strongly interconnected and interdependent: There ought to be no attack that does not also have a suitable defense available nor any defense strategy that does not mitigate a specific attack scenario. This is revealed by our model, since it enforces the linkage between attacks and defenses. All of these findings can be considered to be gaps in literature and, as such, potential areas for further research. In particular, we noticed that information we consider to be helpful to decision-making in the car industry (e.g. the cost of defense mechanisms) was never provided in the literature we analyzed.

5 Conclusion and Future Work

This paper presents a comprehensive model for security analysis in the automotive domain. Our model consists of three submodels: the environment model, illustrating and structuring all relevant assets in the automotive environment; the attack model, describing attacks as capability-exploit chains of attackers to their final objective; and the defense model, which is used to classify different defense strategies. Additionally, we offered a structure for existing knowledge contained in the academic literature on the subject, using this model; and we discussed the examples of four different ways to apply this structure.

We are aware that this paper does not fully elaborate the detailed usage of the model. We intentionally presented the model in an abstract level, and only sketched exemplary applications to keep the underlying structure generally applicable and independent from any concrete example. The particular usage is always specific to the user of the model and the model could be refined as needed. Nevertheless, our model can be used to integrate existing knowledge and enable a structured analysis of alternatives.

Although our mapping of selected research shows the applicability of our model to existing research, the results of this mapping are only preliminary; this being because it is not typical of the whole body of knowledge concerning security in the automotive domain. As future work, we are planning to extend and continue mapping those attacks and vulnerabilities that have already been documented and this research will be published in the future. This will enrich our formalized model with more detailed attack descriptions, with the aim being to enable solid, well-founded discussion of issues in the automotive domain. Furthermore, we plan to build a small web-based navigator that will transform the voluminous literature mapping found in this paper into an explorable knowledgebase that will be available publicly, for researchers and industry experts alike. Finally, we also plan an experimental evaluation of the benefits of our new model, to be carried out in conjunction with experts in this area, drawn from science and industry. While the instantiation already indicates the potential of this model, only further investigations will be able to demonstrate its value in helping these experts improve automotive security.

References

1. Manfred Broy, Ingolf H Kruger, Alexander Pretschner, and Christian Salzmann. “Engineering automotive software”. In: Proceedings of the IEEE 95.2 (2007), pp. 356–373.
2. Riccardo Coppola and Maurizio Morisio. “Connected car: technologies, issues, future trends”. In: ACM Computing Surveys (CSUR) 49.3 (2016), p. 46.
3. Dennis Kengo Oka, Takahiro Furue, Lennart Langenhop, and Tomohiro Nishimura. “Survey of vehicle IoT bluetooth devices”. In: Service-Oriented Computing and Applications (SOCA), 2014 IEEE 7th International Conference on. IEEE. 2014, pp. 260–264.
4. Ning Lu, Nan Cheng, Ning Zhang, Xuemin Shen, et al. “Connected vehicles: Solutions and challenges”. In: IEEE Internet of things journal 1.4 (2014), pp. 289–299.
5. Richard Viereckl, Dietmar Ahlemann, Alex Koster, Evan Hirsh, et al. “Connected car report 2016: Opportunities, risk, and turmoil on the road to autonomous vehicles”. In: PwC, last accessed (15.05. 2018) at: http://www. strategyand. pwc. com/reports/connected-car-2016-study (2016).
6. Christoph Ponikwar, Hans-Joachim Hof, Smriti Gopinath, and Lars Wischhof. “Beyond the Dolev-Yao model: Realistic application-specific attacker models for applications using vehicular communication”. In: arXiv preprint arXiv:1607.08277 (2016).
7. Muhammad Asim, Jorge Guajardo, Sandeep S Kumar, and Pim Tuyls. “Physical unclonable functions and their applications to vehicle system security”. In: Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th. IEEE. 2009, pp. 1–5.
8. Tobias Hoppe, Stefan Kiltz, and Jana Dittmann. “Applying intrusion detection to automotive it-early insights and remaining challenges”. In: Journal of Information Assurance and Security (JIAS) 4.6 (2009), pp. 226–235.
9. Stefan Axelsson. Intrusion detection systems: A survey and taxonomy. Tech. rep. Technical report, 2000.
10. Peyman Kabiri and Ali A Ghorbani. “Research on intrusion detection and response: A survey.” In: IJ Network Security 1.2 (2005), pp. 84–102.
11. Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, et al. “Comprehensive Experimental Analyses of Automotive Attack Surfaces.” In: USENIX Security Symposium. San Francisco. 2011.
12. Ivan Studnia, Vincent Nicomette, Eric Alata, Yves Deswarte, et al. “Survey on security threats and protection mechanisms in embedded automotive networks”. In: Dependable Systems and Networks Workshop (DSN-W), 2013 43rd Annual IEEE/IFIP Conference on. IEEE. 2013, pp. 1–12.
13. Simon Parkinson, Paul Ward, Kyle Wilson, and Jonathan Miller. “Cyber threats facing autonomous and connected vehicles: future challenges”. In: IEEE Transactions on Intelligent Transportation Systems 18.11 (2017), pp. 2898–2915.
14. Keen Security Lab of Tencent. Experimental Security Assessment of BMW Cars: A summary report. https://keenlab.tencent.com/en/2018/05/22/New-CarHacking-Research-by-KeenLab-Experimental-Security-Assessment-of-BMW-Cars/. [Online; acc. 2018-05-15]. 2018.
15. Karl Koscher, Alexei Czeskis, Franziska Roesner, Shwetak Patel, et al. “Experimental security analysis of a modern automobile”. In: Security and Privacy (SP), 2010 IEEE Symposium on. IEEE. 2010, pp. 447–462.
16. Chen Yan, Wenyuan Xu, and Jianhao Liu. “Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle”. In: DEF CON 24 (2016).
17. Dieter Spaar. “Auto, öffne dich!” In: c’t Magazin 5 (2015), pp. 86–89.
18. Min-Joo Kang and Je-Won Kang. “Intrusion detection system using deep neural network for in-vehicle network security”. In: PloS one 11.6 (2016), e0155781.
19. Computest. The connected car – Ways to get unauthorized access and potential implications. https://www.computest.nl/wp-content/uploads/2018/04/connected-car-rapport.pdf. 2018.
20. Jürgen Dürrwang, Johannes Braun, Marcel Rumez, and Reiner Kriesten. “Security Evaluation of an Airbag-ECU by Reusing Threat Modeling Artefacts”. In: (2017).
21. Matan Levi, Yair Allouche, and Aryeh Kontorovich. “Advanced Analytics for Connected Cars Cyber Security”. In: arXiv preprint arXiv:1711.01939 (2017).
22. Keen Security Lab of Tencent. Car Hacking Research: Remote Attack Tesla Motors. https://keenlab.tencent.com/en/2016/09/19/Keen-Security-Lab-of-Tencent-Car-Hacking-Research-Remote-Attack-to-Tesla-Cars/. [Online; acc. 2018-05-15]. 2016.
23. Bill Howard. Hack the diagnostics connector, steal yourself a BMW in 3 minutes. https://www.extremetech.com/extreme/132526-hack-the-diagnostics-connector-steal-yourself-a-bmw-in-3-minutes. [Online; acc. 2018-05-15]. 2012.
24. Troy Hunt. Controlling vehicle features of Nissan LEAFs across the globe via vulnerable APIs. https://www.troyhunt.com/controlling-vehicle-features-of-nissan/. [Online; acc. 2018-05-15]. 2016.
25. Pen Test Partners. Hacking the Mitsubishi Outlander PHEV hybrid. https://www.pentestpartners.com/security-blog/hacking-the-mitsubishi-outlander-phev-hybrid-suv/. [Online; acc. 2018-05-15]. 2016.
26. Stanley Chow, Philip Eisen, Harold Johnson, and Paul C Van Oorschot. “White-box cryptography and an AES implementation”. In: International Workshop on Selected Areas in Cryptography. Springer. 2002, pp. 250–270.