A Privacy-aware Data Access System for Automotive Applications

Christian Plappert, Daniel Zelle, Christoph Krauß
Benjamin Lange, Sebastian Mauthofer
Fraunhofer SIT
Darmstadt, Germany
{christian.plappert, daniel.zelle, christoph.krauss}@sit.fraunhofer.de
{benjamin.lange, sebastian.mauthofer}@sit.fraunhofer.de

Jonas Walter, Bettina Abendroth
IAD, TU Darmstadt
Darmstadt, Germany
{j.walter, abendroth}@iad.tu-darmstadt.de

Rasmus Robrahn
Unabhängiges Landeszentrum für Datenschutz
Kiel, Germany
uld65@datenschutzzentrum.de

Thilo von Pape
Universität Hohenheim
Stuttgart, Germany
thilo.vonpape@uni-hohenheim.de

Hendrik Decke
Volkswagen AG
Wolfsburg, Germany
hendrik.decke@volkswagen.de

***Abstract*—The introduction of Information technology (IT) in modern vehicles enables a plethora of new applications ranging from value added services up to autonomous driving vehicles. However, this also introduces new threats with regard to IT security and privacy. In this paper, we discuss the new privacy issues and propose a privacy-aware data access system for automotive applications. Our system informs the user over all privacy aspects and enables him to control third-party access to his personal data. We developed an easily usable human machine interface (HMI) and an underlying policy system to control data flows which is compliant to the European General Data Protection Regulation (GDPR). Our system can be easily integrated in future automotive architectures.**

I. Introduction

Information Technology (IT) is one of the main drivers for innovation in modern vehicles and of paramount importance towards autonomous vehicles. Up to 100 Electronic Control Units (ECUs) realize different vehicular functions in hardware and software. ECUs communicate with each other via different bus systems such as CAN, LIN, MOST, FlexRay, or automotive Ethernet. In addition, vehicles become more connected to the outside world, e.g., Vehicle to Vehicle communication (V2V) or even direct connections to the Internet via a connected smartphone or an integrated telematics ECU. The Internet connectivity enables applications such as music streaming, web browsing but also the execution of vehicle functions such as unlocking the car or starting / stopping the engine in car sharing scenarios. The plethora of available data also enables new business models, making the connected vehicle more and more interesting for third parties. For example, insurance companies such as Metromile [1] or Allstate [2] offer so called pay-as-you-drive (PAYD) insurance tariffs where the vehicle owner is charged based on driving behavior.

However, the increased use of IT also introduces new threats in terms of data security and data protection. Successful attacks may have serious consequences and vary from monetary threats (e.g., odometer manipulation) over privacy leaks (e.g., generation of movement profiles or the analysis of the driving style) to threats to the life and limb of passengers.

In this paper, we focus on the privacy aspect and present an approach to analyze and control information flow of vehicle data. We propose a privacy monitoring system for vehicles that informs the user about sensitive data flows and enables him to control these data flows. Furthermore, we use Privacy Enhancing Technologies (PETs) to modify data according to user settings. For this purpose, we developed an HMI which allows the user to employ his own privacy settings (with his desired level of privacy protection) and informs the user about information flows and all privacy-related data. The HMI interacts with our newly developed policy architecture which monitors and controls the data flows of the vehicle. Central design targets of our system are high usability, i.e., all information and control functions must be easily understandable, and fulfillment of the relevant legal aspects, i.e., compliant to the European General Data Protection Regulation (GDPR). In addition, our system can be easily integrated in future automotive architectures.

The remainder of the paper is structured as follows: Section II describes the general setting we assume, including a reference architecture and relevant use cases. The legal and user requirements are elaborated in Section III. Our Privacy-aware data access system is presented in Section IV and Section V respectively. The former describes our privacy HMI while the latter describes the underlying architecture layer enforcing the functionality of our privacy HMI. Finally, Section VI presents related work while Section VII concludes the paper and gives an outlook for further work.

II. Setting

In the following, we discuss the reference architecture and relevant use cases we assume for our privacy-aware data access system.

A. Reference architecture

To realize an actually implementable solution for the varying vehicular architectures of all the different manufacturers, we developed an abstract reference architecture. The focus of this reference architecture is on describing the relevant data flows and interfaces regarding privacy and to generalize between the solutions of different manufacturers. Therefore, we do not differentiate between different electronic control units but concentrate only on general functionalities realized by creating, transmitting, or using data.

As depicted in Figure 1, we focus on the various interfaces, where data may leave the car. These interfaces can be roughly categorized in three classes. First, we added the on-board diagnostics (OBD)-port, which is prescribed by law and can be used to access a varying amount of data from the car's electronic control units. Second, we see a large and growing amount of wireless interfaces such as Bluetooth, WiFi, Cellular, or Car2Car-Communication. As these interfaces do fulfill specific roles and cannot be freely used for different functions, they should be listed individually. But regarding privacy they can be combined and only be specifically addressed in the different use cases. Third and last, we have physical ports, which are mainly used to insert data into the car but may also be used to extract data from the car. This includes CD/DVD slots, USB or SD ports.

Inside the car, data is transferred between different components, electronic control units and systems. Additionally, the car is partitioned into different logical domains like *Powertrain* or *Driving Dynamics*. Within these domains, we see a variety of electronic control units, actors and sensors, which are connected by one or multiple communication channels. The communication channels of the different domains are separated from each other by a central gateway, which controls the data transferred between the domains. Further, there are some particular components like the instrument cluster, an infotainment system or central control units for functions across several domains.

In addition to the vehicular components, we introduce further relevant components and actors. These include a car mechanic reading from the diagnosis port, further road users, mobile phones and computers, which communicate over different interfaces with the car. Most importantly we add the *Internet* as a component, including different backend servers and a data market place. The backend servers are differentiated between the manufacturer's backend and third party servers. In front of the various backend components is an optional telecommunication anonymisation service, which mainly conceals the car's IP address. The data market place is a fictional place where different manufacturers and third parties may exchange data at a central place.

Finally, the most important actor is the user. The user may interact with the HMIs, the infotainment system of the car, the mobile phone, the computers, or other actors of the reference architecture. Which components the user interacts with is then highly dependent on the use case under consideration.

Table I
USE CASE OVERVIEW

Category	*Use case*
Multiple car usage	Car Sharing, Garage Service
Location-based services	React to current position
Third party and smartphone integration	Android Auto, Paket-Auto
Statistical analysis	Environment, wear analyses
Electro mobility	Charge and pay
Passenger monitoring	Driver behavior, driver monitoring

B. Use cases

To create a general privacy-aware data access system, it is important to regard all relevant use cases. We consider ten use cases in six distinct categories which cover, in our opinion, all relevant aspects of data usage with regard to privacy.

Table I shows an overview of the use case categories and considered use cases within the respective category. The first category *multiple car usage* includes uses cases, where the car is used by more than one person including workshops or car sharing. The second category *Location-based services* includes a use case, where the car or services of the infotainment system react to the current position of the car. The third category focuses on *third party and smartphone integration* through the examples of Android Auto and parcel delivery into the car. *Statistical analysis* of car data is addressed in the fourth category, where environment or wear data is analyzed. Payment and the charging of an electronic car is examined in *electromobility*. The last category *passenger monitoring* inspects use cases where the driving behavior or general state of the driver and passengers are recorded and submitted.

Using these example use cases, our privacy-aware data access system was developed regarding the legal and user requirements described in the following section.

III. REQUIREMENTS TO A PRIVACY PRESERVING SYSTEM

A privacy preserving system needs to fulfill multiple requirements. Two central aspects are compliance to relevant laws and the requirements of users with regard to usability of the system. The legal and user requirements are described in the next sections.

A. Legal Requirements

The General Data Protection Regulation (GDPR), which will apply from the 25th of May 2018, is the legal framework for the development of solutions for the connected car, that have to comply with data protection requirements because personal data is processed. The measures that have to be taken can be systematized based on protection goals. Seven fundamental protection goals have been identified, namely data minimization, availability, integrity, confidentiality, unlinkability, transparency, and intervenability [3]. Here, a short introduction to the rules of the GDPR which demand transparency and intervenability for the data subject (a person which is identified or identifiable based on the processed personal data) will be given. They have to be operationalized in systems and

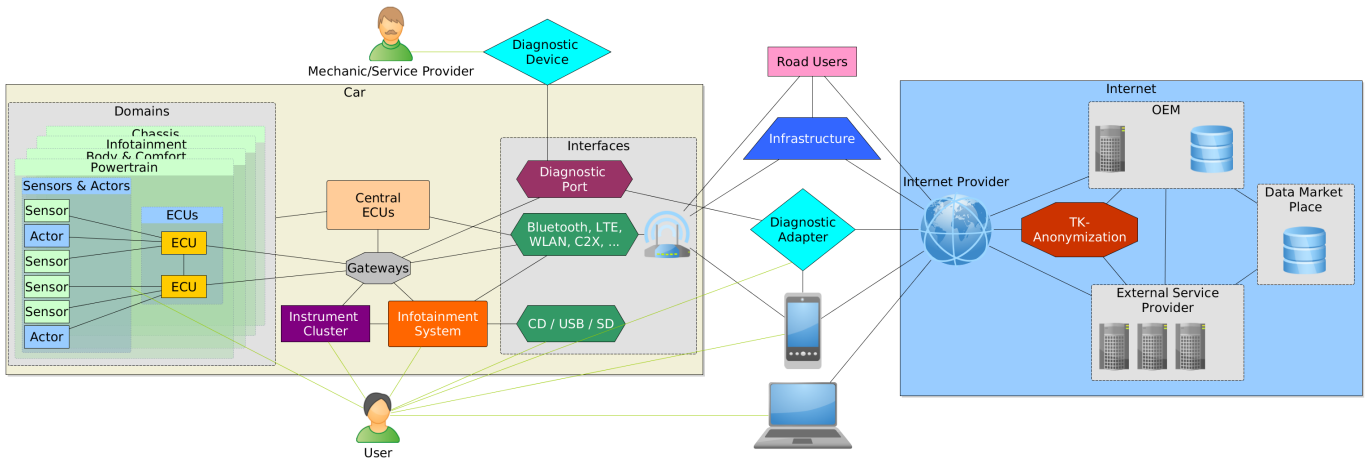


Figure 1. Connected Car Reference Architecture

processes that have to fulfill the requirements of the GDPR, such as the processing of personal data of connected cars.

1) Intervenability: The data subject has several rights which allow him to intervene the processing. According to Article 16 GDPR the data subject has the right to rectification. Inaccurate data has to be rectified; incomplete data has to be completed. This right is especially important for the data subject, when he depends on the accuracy and completeness of the data. Examples could be pay-as-you-drive insurance models or data that could be used as proof to determine guilt or innocence, when a car accident is investigated in court.

Article 17 GDPR gives the data subject a right to erasure, which is also known as the much discussed right to be forgotten. Personal data that concerns a certain data subject has to be erased under certain preconditions, e.g., if the personal data is no longer necessary, if the data subject withdraws consent or if personal data have been unlawfully processed.

The right to restriction of processing is useful for the data subject in cases, where the accuracy of the personal data is contested, the processing is unlawful, but the data subject opposes the erasure or when personal data is no longer needed for the purposes of the processing but are required by the data subject for legal claims. Restricting the processing of personal data is defined in Article 4 (3) GDPR as marking stored personal data with aimed at limiting their processing.

Article 20 GDPR gives the data subject a right to data portability, aiming to prevent user-lock-in situations, where the data subject does not switch service providers because he is unable to transfer his personal data. The right to data portability allows the data subject under certain preconditions to transfer his personal data from one controller to another in a structured, commonly used and machine-readable format. The right to object is found in Article 21 GDPR. The data subject can object processing activities based on certain legal grounds and the data controller can only continue the processing of personal data if he cannot provide compelling legitimate grounds for the processing which override the interests, rights and freedoms of the data subject or for legal claims.

2) Transparency: Article 5 (1) (a) GDPR states that personal data shall be processed in a transparent manner in relation to the data subject. Special regulations regarding transparency can be found in the Article 12 to 15 GDPR. The data controller has to take appropriate measures to provide any information in a concise, transparent, intelligible and easily accessible form, using clear and plain language. If appropriate, the information shall be given by electronic means. In the case of connected cars it is appropriate to use the HMI to display information concerning personal data because this is the option that is closest to the actual data processing from the point of view of the data subject compared to other options like the operating manual. Also, while the data processing activities could change, the operating manual will stay the same. The requirements regarding the use of clear and plain language aim to enable the data subject to not only be formally informed but to also understand the given information. For the data controller, this is a challenging task, as he has to transfer complex technical systems into information that is understandable for laymen with different backgrounds. Multi-layered information [4] or privacy icons [5] are a good solution for such cases. The task becomes even more challenging, when it is considered, which content the information must have according to Article 13 GDPR. Not only is the data subject to be informed about identity and contact details of the controller, the recipients of personal data, and the purposes of the data processing, but also about the period the personal data will be stored for, his rights to request access, rectification and erasure of personal data and his right to lodge a complaint with a supervisory authority.

B. User Requirements

The requirements posed by the EU-GDPR involve various preconditions regarding the users' perceptions and behavior with respect to the protection of their data. Namely, the condition of explicit valid consent for both the collection of data and their purposeful use can only be fulfilled under the conditions that the users are aware of the collection of the data and the purposes by which this collection is justified.

Further, the requirement for users to be able to withdraw their consent and to request erasure of data relating to themselves necessitates that users be able to control the disclosure of their data at any time, and not just occasionally during the process of installing apps. These preconditions of user awareness and control depend in themselves on factors inherent in the technology, in the users and in the situations of usage and control. To identify these requirements in the context of connected cars, we conducted various studies with drivers and users of connected cars.

The following requirements are derived from guided interviews with 17 drivers of connected cars [6], a representative survey among German car drivers and an online survey among consumers [7].

First, in order to allow users to make self-determined decisions about using certain services in the first place, possible data flows need to be transparent to the user. This includes not only the data types themselves but also, i.a., information about the data receiver, purpose of data collection and storage period. The requirement of transparency goes beyond the principle that the information must be somehow accessible for technically savvy users, demanding rather that it is actively elaborated in a form understandable for all users in the given situational contexts, e.g., through easy to grasp iconographic visualizations of data types. In general, information should be presented when they become relevant and the user's perception is not restricted, e.g., before the decision to allow certain data flows. However, to some extent certain information can even be displayed in driving situations without distracting the driver from the driving task, e.g., real time display of apps requesting certain data types.

Second, while the transparency requirement makes data collection transparent to the user, he needs to make self-determined decisions based on this information. In other words, within this requirement the user must be able to control all relevant data flows that are not enforced by law (e.g., in case of OBD-II data). This includes also the control of data after collection, e.g., the possibility to delete collected data from data receivers. Moreover, it must be ensured that control over specific user data can only be made by the user himself and not by other entities accessing the car, e.g., workshops or the vehicle owner in case he is not driving himself. A prerequisite for control is to give users alternative options for using services. Since today's commonly used all-or-nothing-approaches where users only have the choice to use certain services while sharing all their data or not being able to use the service at all are of limited value because users may depend on the offered functionalities[8], services should offer options where users can select more granularly, e.g., by disabling certain features they are not interested in. Finally, all control options should be both easy to understand for the majority of users with different background knowledge and they should adapt to the situations they are used in, e.g., full control when the car is parked and only reduced visual information during driving.

Third, our studies show that users primarily perceive the services' functionality and user interface while privacy issues are only secondary. Therefore, our system must preserve the services' core functionality and user experience while simultaneously achieving both transparency and control requirements. Thus, users ideally do not have to trade-off functionality and user experience against control over privacy.

Fourth, it is important to account for the unavoidable limitations of any user-centered approach to data-protection which result from the limitations to the users' competencies as well as situational restrictions [9]. Therefore, the technology must provide a solid level of data protection that is independent of any user-sided regulation on the basis of the principles of privacy-by-design and privacy-by-default.

IV. Human Machine Interface

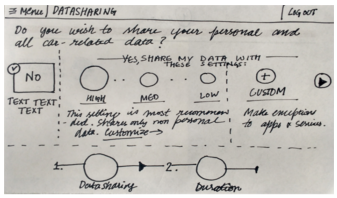
To provide vehicle passengers an effective means to control their privacy in connected vehicles in a self-determined manner, a vehicular user interface was developed. Resulting from the specified user requirements, the explicit goals of this vehicular privacy application were ...

- to enable informed decision when downloading, updating or interacting with a third party service.
- to enable the user to control his own privacy settings.
- to facilitate the understanding of privacy relevant processes.
- to guide the user when making privacy settings.
- to apply for all vehicular services and application that run directly on the car.

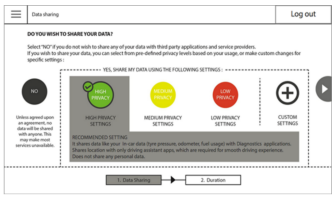
Based on the human centered design process (ISO 9241-210 [10]), an iterative approach was chosen that focused on an early and regular integration of user feedback in multiple steps of the development. In the following section, the procedure is detailed and then the development of the interface over time is depicted.

A. Procedure

ISO 9241-210 suggests an iterative process to design systems and products in a human centered manner. Following this process, we first defined the goals presented above. To reach these goals, we derived the above detailed requirements from legal, technical and user-centric points of view. Subsequently, creative techniques like brainstorming and mind mapping were used to associate content with the requirements. In order to structure, label, and organize the content, we laid out a first information architecture and applied rapid prototyping methods (e.g., [11]). In course of multiple short runs of small usability tests (with n ranging from 4 to 8), a user interface was developed. Starting from rapid low-fidelity paper prototypes, the complexity and quality of the prototypes increased with each iteration. In course of these iterations, one paper-prototype and three digital wireframes were created. Each iteration involved the derivation of design specifications from requirements and/or feedback from previous user tests. According to these specifications, the current prototype of that stage was (re-)designed and updated. The newly updated



(a) Paper prototype.



(b) Click dummy.

(c) Dynamic layout (first iteration).

(d) Dynamic layout (current design).

Figure 2. History of interface development.

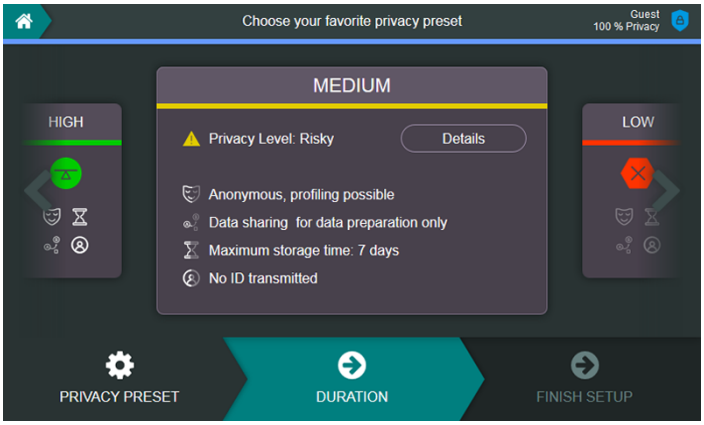


Figure 3. Selection of privacy policies.

prototype was then used for small usability tests from which concrete user hints were gained.

Figure 2 provides a rough overview over the development that the interface layout took. Starting with very simple paper prototypes, the focus was first put on content and information architecture (2a). Subsequently, a first click dummy was created to simulate interactivity (2b). In course of the following iterations, more complex click dummies arose, resulting in the current dynamic layout (2c and 2d).

B. Structure

The predominant goal of the user interface is to enable the user to make informed privacy relevant decisions within the vehicular context. To do so, transparency of data flows and controllability of data disclosure have to be increased significantly in comparison to the current state in modern cars. However, deciding on privacy settings is a complex task that requires a high level of attention. Though we tried to simplify settings as much as possible without losing a necessary profundity, the privacy task is likely to conflict with requirements of attentional resources given by the driving situation. Following the suggestions of NHTSA (National Highway Traffic Safety Administration) [12], we decided to only allow complex settings when the car is parked, while specific screens are available for the driving situation.

C. Screens

For the sake of simplicity, the case of the first usage of the application is sketched. Since a user might register a profile and decides on his preferred privacy policy in course of the first usage, a broad span of the application is covered.

First, the user can choose to register a profile for the application or use the application as a guest. By using profiles, the own privacy settings can be saved, such that privacy settings can be accessed in different vehicles or hand-held devices whenever it is desired. Moreover, profiles allow multiple users to use the application within a car while access to a profiles privacy settings is restricted to authorized persons only. However, if no registration is desired or not suitable (as it might be the case in, e.g., car rental), the application can be used as "guest". For users in a guest status, all functions are available, but settings are only valid for one ride and cannot be saved.

If the user logs in for the first time or logs in as a guest, he is invited to take an explanatory tour before diving into the settings. The tour presents the purpose of the vehicular privacy application using explanatory graphic elements and short text snippets on six slides (s. Figure 3 for an example tour screen).

After having completed the tour, the actual definition of the privacy policy begins. To simplify the privacy decision, a set of predefined privacy policies is suggested. Each privacy policy is described by detailing the degree of anonymization, the involvement of third parties, the maximal duration of data storage, the category specific percentage of application being available and information on the transfer of sensitive data types (s. Figure 3). There are four predefined privacy policies, ranging from no data transfer at all to a liberate privacy setting. Moreover, the user has the option to define a customized privacy policy. In the following, both variants (selecting a predefined policy or customize ones one policy) are detailed.

1) Selection of a predefined policy: If the user selects a predefined privacy policy, he subsequently decides on the duration of the validity of the selected policy. There are several options available, ranging from a temporal restriction to a single ride to an infinite validity. If an option other than infinite validity is chosen, the user receives a reminder of his policy settings as soon as the car is started for the first time after the previously defined validity interval ran out.

2) Customize a policy: If the user decides to customize an own privacy policy, a multi-step process is started. First, in analogy to the initial privacy selection step, the user is presented with four different basic privacy settings from which the customization starts. Again, the basic privacy settings range from no data transfer to a liberate privacy handling. After having selected one of these settings, more advanced configurations can be made (s. Figure 4). The user can set allowances for single data types (e.g., only position data), filter apps for specific functions (e.g., only safety-enhancing functions) or select specific services. Moreover, detailed infor-

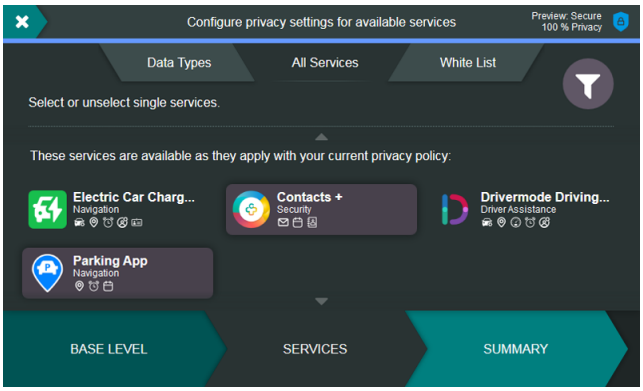


Figure 4. Customization of privacy policies.

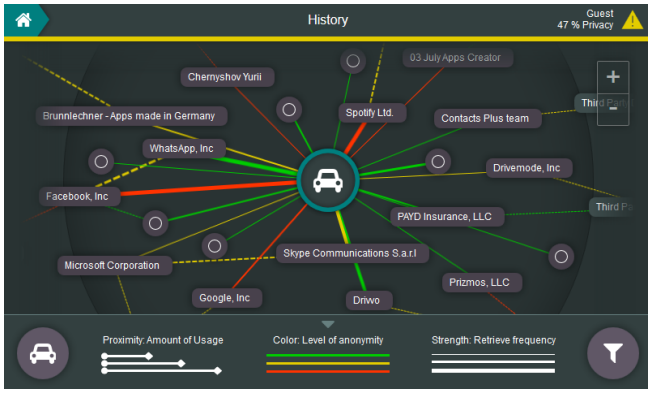


Figure 5. History graph.

mation for each service are available. Here, information about which data is used for which function and the identity of the data receiving parties are available. Also some service-specific settings can be made if multiple privacy options are provided by the service.

Summarized, these steps enable users to define an own global privacy policy. However, if a user wants to set exceptions for single services, a white-listing tool is available. Here, single services can be defined for which the customized policy does not apply. After having set a custom privacy policy, the policy can be saved. Subsequently, the user proceeds to the duration setting that has been described above and hence finalize the policy definition. Once a customize policy has been finalized, it is linked to the user account such that it is available in each vehicle in which the respective account is activated.

Once a policy has been defined or selected, it is applied on all services that are installed on the vehicle. To enhance transparency of the ongoing data transfer, a history graph is available. Here, all data transfers to all receiving parties and related third parties are visualized. As shown in Figure 5, there are three parameters used to communicate the characteristics of the data flow. First, the distance of a data receiving party to the own car displays the frequency of use of the services offered by the respective data receiving party. Second, the thickness of a connection indicates the frequency of transfer requests. Third, the color of the connection symbolizes the quality of anonymization.

All the above mentioned options are only available when the car is parked. During driving, the currently selected privacy policy is permanently applied. However, within feedback screens in case of special events, exceptions for the current ride can be defined. The driver is provided with an informative prompt in case of service-related events that might change the data requests of a service (new installation or update) or user-triggered events that conflict with the current privacy policy (access to a service or a subfunction of a service that is currently blocked due to privacy reasons). Despite of a carefully reduced design and parsimonious selection of only few interaction screens during driving, there is no pre-defined limitation of interaction screens that can pop up during a ride. However, straight-forward instruction on these screens guide the user towards a compliant usage of the application also during a ride. Moreover, in multiple user tests (overall $N > 20$), no subject tried to open restricted apps or functions repeatedly such that it can be assumed that there is a negligible risk of distraction caused by the current number of screens during a ride.

In our privacy application, all privacy policy definitions are restricted to the parked state of the car though a front passenger could have taken over the settings during a ride as well. We decided to do so as false alarms in front passenger detection (e.g., little children or heavy boxes on the front passenger seat) might enable the driver to engage in cognitive demanding settings which might be critical for the driving task. Anticipating this potential attentional conflict, we allowed access to all cognitive demanding settings only if the car is parked.

Thus, the vehicular privacy application offers comprehensive control on privacy while ensuring a high level of transparency. Users can actively decide on their own privacy policies and are provided with the possibility to monitor the data flows of their currently used services. With the combination of an intuitive guidance as well as advanced privacy settings, there are adequate settings for both, novice as well as expert users, available.

V. Privacy-Aware Policy System

The implementation of both the specified requirements and the user's individual privacy settings adjusted within the HMI is enforced by the underlying privacy-aware data access system which uses policies to control sensitive data flows.

In general, it is designed as a "privacy firewall" between the car and the car's environment, whereas the environment is defined as the point where data control is lost to external parties, e.g., apps installed on the car's infotainment system or diagnostic devices connected to the OBD port. For brevity we refer to them as "external services" which either have an additional backend component with which data can be exchanged or provide direct insight to external parties.

It has to be noted that in some cases it is not allowed to restrict the vehicular communication with external services,

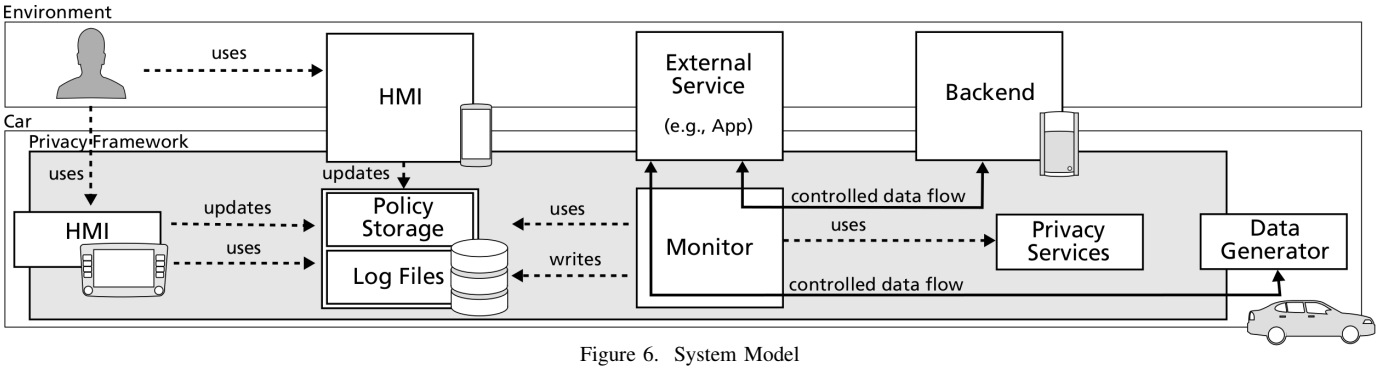


Figure 6. System Model

e.g., in case of legal requirements. In these cases, the focus of the project is to provide transparency to the user.

A. System Model

As depicted in Figure 6, our basic system consists of the user with his privacy HMIs, an external service with its respective backend component, the car's "data generators", e.g., sensors or electronic control units (ECUs), and our developed privacy framework. The privacy framework has interfaces to the HMI, the external service with its backend and to the data generators. It can be further subdivided into a storage component consisting of both a policy storage and log files, a monitor component which observes and controls data flows and various privacy services which, i.a., implement PETs and are used to carry out encrypted communication to the service's backend.

As described in Section IV, the HMI enables control and transparency features by (i) informing the user about privacy-sensitive data flows and (ii) letting him control them by adjusting privacy settings. More specific, transparency is enabled by storing relevant data requests and the meta-data of all data flows leaving the car (and the content if required). This information can be accessed by the HMI to display it in a suitable way to the user, e.g., in form of a transparency report. The control of the data flows is enabled by transforming the user's settings as well as legal and user requirements into machine-interpretable policies which are stored in the policy storage. While user related settings and requirements can be viewed and adjusted via HMI, static requirements (e.g., legal requirements) can only be viewed but not adjusted by the user.

Based on the stored policies, the monitor component decides how to handle specific data flows to the environment. The decision process can be further enriched with meta data which is, i.a., retrieved from the log files, e.g., to get knowledge about the interval of data flows in the past.

Depending on the decision of the monitor component, certain data can be delegated to various privacy services, e.g., in order to coarsen the data before sending it back to the requesting service or to its backend.

B. System Framework and Policies

Generally, there are two relevant kinds of external services that want to obtain data from the car: (i) services that do not have a backend component and process data locally and (ii) services that rely on a backend component to work and to which they need to send the car's data to. While the first variant is uncritical since data processing is done only locally, in the second variant potentially privacy-sensitive data is leaving the car. For the latter data flows the user must at least be informed about and should be able to control the data flow.

Therefore, our privacy-aware data access system is instantiated as a framework that needs to be implemented by every external service that wants to get data from the car. The framework enforces strict communication interfaces that are used to establish communication both with the car as well as with the car's environment. Among other things, the framework also carries out encrypted communication with backend components to avoid external services to undermine our monitoring by encrypting data themselves.

By enforcing the framework to every external service, our system is able to monitor all data flows and pre-process data before the service or the backend component receives it. In order to mitigate scenarios where an adversary tries to circumvent the system's data access control by hiding sensitive data in seemingly benign data structures, the framework additionally enforces the usage of predefined data structures for communicating with backend components. The payload of these data structures is strictly defined and to a certain extent verifiable by the system.

The data flow control is done based on the policies stored in the policy storage component and which come in two flavors: (i) static policies based on legal or general user requirements, e.g., usability requirements in the automotive context and (ii) dynamic user-defined policies. Whereas, the first are deeply-rooted within the system and can only be changed via system update, e.g., when laws change, the latter policies are created based on the user's settings in the HMI. Conflicts arising between user- and static policies are always decided in favor of static policies. However, even when laws enforce data transmission, our framework still logs these data flows and informs the user accordingly.

C. Data Processing

Prior to first usage of a service, e.g., before installation, the user is preferably informed by the service about requested data and presented with different privacy options and their

consequences to choose from. This is accomplished by developers writing respective policies with easy-to-understand meta information displayed to the user via HMI. Besides the selectable options, the policies contain certain meta data, i.a., information about the data requester like headquarter location, e.g., for deriving a privacy level at the data requester, requested data and data type as well as the purpose for information collection which if not present would make the request illegal. Additionally, policies can refer to the stored data of potential backend communication partners, e.g., IP addresses and certificates used to establish secure communication. Moreover, each requester is associated with a certain knowledge domain which allows on the one hand to reconstruct data pools across multiple seemingly separate requesters and on the other hand can, e.g., identify different business units of the same company where knowledge is not shared among. Thus, the concept of the knowledge domain enables more transparent data flows both within the same company as well as across different companies.

The user now can decide for a suggested policy template or edit a suggested one which is afterwards stored in the policy storage. Afterwards, the service can start communicating with the car or is only now installed.

As soon as the service exchanges data with the car there are two distinct data flows: (i) service-to-car communication to retrieve car data and (ii) service-to-backend communication, e.g., for forwarding gathered data to a remote server. Regardless of the type, our framework enforces that each data flow is initiated by a service request to the monitor component to either get data from the car or to send data to the respective backend component. By analyzing the service's requests, generally blocked data types, e.g., GPS location, can be already handled by the monitor without communicating with the car's data generators. Besides analyzing the request and possibly controlling the resulting data flow via policies, the monitor stores each request in log files which can be used by the HMI to obtain a transparency report of all data flows including possible destinations.

The monitor then matches each request with the policies from the policy storage including stored meta data and calculates the result. In order to handle additional constraints not derivable from the request and meta data themselves, e.g., user-defined time constraints where a certain data type is only allowed to be retrieved again after a certain time period, further environment data can be requested, e.g., from the log files where the last requests are listed.

Depending on the processing result of the policy, the request is either denied or permitted. Optionally, depending on the underlying policy, a permission can be restricted by an obligation which requires the fulfillment of a pre-condition, e.g., calculating an average over a set values from the requested data type, before sending the by then modified response data back. Equally to simple unique data requests, our system can handle publish-subscribe requests where a service wants to get data for a self-defined interval in a self-defined time period, e.g., for wear analyses.

Anytime during the runtime of the system, the user can update the policies with the respective HMI. By the system's design, unique data requests are by then already validated against the new policy. However, all publish-subscribe requests affected by the policy change need to be manually terminated and according apps need to be initiated to renew their subscription-request which then will also be validated against the new policy.

Updating the policy affects not only newly requested but also previously gathered data that could be even already exported to external parties. Despite exported data being out of direct control of our policy system, meta information attached to all data send to external parties allows them to apply the same policies as inside the car and update them accordingly. As soon as an updated policy is recognized either by car or in the backend all data that is affected by the policy change need to be processed again. Depending on the supported privacy options, the service can decide if the new data still constitutes an acceptable level of abstraction for further processing or will inform the user to weaken the privacy setting again.

D. Privacy Services

Depending on obligations in policies it might be necessary to fulfill specific preconditions before transmitting data. A number of Privacy Services are designed to fulfill these conditions using PETs. We use four different classes of PETs:

1) Data anonymization: Sensitive data might have the constrain to be anonymized or pseudonymized before transmitting it to the environment. A very simple form of anonymization/pseudonymization is to delete identifying informations such as names, VIN, serial numbers, etc. or – in case of pseudonymization – to replace them by pseudo-random numbers used as pseudonyms in a series of transmissions. More advanced technologies as surveyed in [13] are necessary for anonymization of GPS-data, driving speed, steering data or specific environment conditions:

- *Aggregation:* Several datatypes are aggregated by calculating average values such as score values of driving styles (used by, e.g., insurance companies). Another possibility is to aggregate a series of events (e.g., detection of fee parking spaces) can be aggregated into a single counting value instead of multiple transmissions.
- *Generalization:* Single GPS-positions or routes are generalized to larger areas such as city areas or by coarsening data with a predefined scale values. The range of generalization depends on the data-specific usage requirements.
- *Statistical perturbation:* By using techniques of data perturbation [13, 14] statistical noise is added to numerical values to distort exact data.
- *Transmission delay:* Using statistical chosen time lags can be used to prevent analyses of data (e.g., a series position data revealing driving routes) and distort correlations that might reveal sensitive information.

2) Authentication: We use two different types of signatures for authentication: Standard signatures as well as group signatures. Both kinds of signatures can be used to prove and verify

the authenticity of sensitive data. While standard signatures simultaneously reveal the identity of the signatory and allow a linkage of different messages of the same signatory, group signatures can be used for anonymous or pseudonymous authentication [15–17]. More advanced techniques include Zero-Knowledge Signatures [18] or Direct Anonymous Attestation [19]. For a predefined group this kind of signature only reveals that the signatory is a legitimate group member. In our context, a group can consist of all members of an external service, a specific set of apps or even a small group of different car drivers. We use standard signatures for securing data containing the identity of the user, e.g., insurance data, and group signatures for transmitting anonymous data, e.g., anonymous statistical data for wear analyses transmitted to car manufacturer.

3) Encryption: Encryption is used to ensure the confidentiality of transmitted data and as building block for Access Control (see below). Encryption of messages is performed by using standard encryption schemes and is used for securing transmitted data, e.g., in establishing encrypted communication with backend components.

4) Access Control: Methods for access control are used for managing, issuing, revoking or verifying permissions to access data stored in the car. We use simple techniques of cryptographic access control (CAS), since they can be integrated in our system without requiring extensive changes to the architecture [20, 21].

E. Implementation

For demonstration purposes and to evaluate the usability studies, we headed for a two-staged implementation approach: (i) simulation via a virtual car server that either generates artificial or respectively replays real car data and (ii) instantiating our system within a full-fledged car (VW Passat) where we can evaluate our system on real ECUs. The HMI is either displayed on a 10.1 inch Android tablet (Galaxy Tab 2) for the first demonstrator or directly on the car's headunit, an 8 inch multicolored touch display with a screen resolution of 800x480 pixels, for the second approach. The corresponding ECU is powered by a 4-core Intel Atom processor running at $1.9\,\mathrm{GHz}$ with $8\,\mathrm{GB}$ of RAM and $32\,\mathrm{GB}$ of storage. For the backend server we used an Intel NUC [22] with attached monitor.

From a software perspective, we implemented our system using Java as programming language, balana [23] as XACML [24] open source Java library for policy description and processing and VW's EXLAP [25] protocol for communication with both the car simulation server and the car. Moreover, all entities from the environment like in-car HMI, external services as well as the backend server are also implemented using Java. The second HMI running on the mobile device is programmed in Android.

Following the XACML standard data flow model, we split our monitor component into submodules, namely Policy Enforcement Point (PEP), Policy Decision Point (PDP), Policy Information Point (PDP), Policy Administration Point (PAP) and Handler. Thereby the PAP is used by both HMIs to configure policies. Incoming data requests from the services – either to allow requesting data from the data generator or to allow sending to the backend – are then intercepted by the PEP. This requests use a simple JavaScript Object Notation (JSON) [26] data structure to transfer either the desired data type and a possible frequency in which they want to obtain data or the payload they want to transmit to the backend. Moreover, these requests contain certain meta data like purpose of data collection, further information about data requester or data receiver, e.g., geolocation or IP.

In either case, the PEP forwards the requests to the Handler which uses a parser to generate corresponding XACML requests which are then send to the PDP which validates them against stored XACML policies and returns a decision. To come to a decision PDP may contact PDP to get further information, e.g., last data request. Depending on the final decision – permit or deny – either an obligation or an advice is attached to the result. In case of deny an advice notifies the service why the request was not allowed, while in case of permit an obligation might be used to further modify responded data according to the policy. In this case data is modified according to the specific obligation in the policy by using our privacy services.

VI. Related Work

A framework combining security and privacy issues in automotive telematics is presented in [27]. This framework includes policies, aggregation of data, integration of apps, control of data flow in the car and user consent. No special attention, however, is given to legal requirements, user requirements or details about PETs that can be used. [28] propose a secure multi-application platform for vehicle telematics. They offer a security architecture with policy management, authentication and access control. Because of their focus in security, privacy issues are not addressed. Privacy and security issues in smart vehicles using temporary pseudonyms without a fundamental architecture are addressed in [29].

In CONVERGE (COmmunication Network VEhicle Road Global Extension) [30] vehicle connection to service providers was investigated. In this context PETs, like proper pseudonymisation techniques with changing pseudonyms, were developed. In general a privacy protection gateway was introduced to the architecture allowing network operators to forward data to service providers pseudonymized or anonymous. This approach however does not imply a control for or information to drivers who deliver these data.

The PRECIOSA project [31] developed privacy protection metrics, concepts and technologies, specifically for location related data in the context of co-operative V2V and V2I systems. However, due to the specific focus, further privacy related application data and the influence of internet-based services are not handled.

The OVERSEE-Project [32] suggests the separation of trust worthy and non-trust worthy application by secure runtime environments that also regulates the access to information. All

information flows, inside and outsite of the vehicle need to use a secure interface, which works on application and user specific rules.

Lately the CarData platform [33] was presented that allows costumers to give third parties access to data form vehicles. Furthermore it allows the user to get an copy of all transmitted data. It furthermore allows the user to identify the data types that re transmitted to third parties. However it does not introduce a separate between applications that use data only when necessary or that constantly request data. Also information flows protected with PETs are not expected. The main strength of this platform approach is also a weakness, it allows the company running the platform to enforce the access control for the user but it give this company complete transparency of all information flows.

A currently running standardization approach, ISO 20078 (Extended Vehicle) [34], describes the connections of vehicles with service providers. This access can be implemented on an OBD adapter (Standardized Adapter), a standardized gateway with cellular connection or OEM-specific servers. A customer portal enables the user to control access to private data. Moreover the standard defines security mechanisms, like OAUTH 2.0 and https, to protect data transmission.

VII. Conclusion

In our work we followed an interdisciplinary approach to allow vehicle users to make informed and effective decisions on data protection relevant information sharing in vehicles. An analysis of data protection laws is accompanied by usability studies and a technical analysis of vehicular architectures. Based on the results, requirements were defined. A reference architecture describes relevant components within the vehicles and outside the vehicles and how they are related and connected. A set of specific use cases, covering the most relevant situations where data protection plays a role in connected vehicles, serve as a basis for the conceptual and technical design of a privacy-aware data access system. The system ensures that the vehicle users have knowledge, understanding and control over which data protection relevant information is transmitted by the connected applications of their vehicles. A fundamental principle of the design is simple usability. The interface ensures that the users can intuitively select fitting privacy settings and the technical system takes care of the enforcement.

As the fine granular privacy adjustments enabled by the presented design are a strong contrast to the wide spread „all-or-nothing “mentality and as they affect the options of the OEMs to provide value-added services, a standardization of the privacy preserving mechanisms might be helpful for faster adoption. Further, as we expect more and more car sharing (private and commercial) in future, the next generation of vehicles should consider the resulting privacy issues to keep up with this trend.

Acknowledgment

The work presented in this paper has been partly funded by the German Federal Ministry of Education and Research (BMBF) under the project ”SeDaFa: Selbstdatenschutz im vernetzten Fahrzeug” [35].

References

[1] Metromile Inc., *Don’t drive much? You can save $500*, 2017. [Online]. Available: https://www.metromile.com/.

[2] Allstate Insurance Company, *We’ve been protecting families for over 80 years.* 2017. [Online]. Available: https://www.allstate.com/.

[3] ULD, *The Standard Data Protection Model*, Mar. 2017. [Online]. Available: https : / / www . datenschutzzentrum . de / uploads / SDM - Methodology_V1_EN1.pdf.

[4] Data Protection Working Party, *Opinion 10/2004 on More Harmonised Information Provisions*, Nov. 2004. [Online]. Available: http://ec.europa.eu/justice/policies/privacy/docs/wpdocs/2004/wp100_en.pdf.

[5] L.-E. Holtz, H. Zwingelberg, and M. Hansen, “Privacy policy icons”, in *Privacy and Identity Management for Life*, J. Camenisch, S. Fischer-Hübner, and K. Rannenberg, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 279–285, ISBN: 978-3-642-20317-6. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-20317-6_15.

[6] SeDaFa Konsortium, *SeDaFa Deliverable D1: Anforderungsanalyse für den Selbstdatenschutz im vernetzten Fahrzeug*, Jun. 2017 (to appear).

[7] C. Krauß, T. von Pape, R. Robrahn, *et al.*, “Selbstdatenschutz im vernetzten Fahrzeug”, *Datenschutz und Datensicherheit - DuD*, vol. 41, no. 4, pp. 217–222, 2017, ISSN: 1862-2607. DOI: 10.1007/s11623-017-0761-8. [Online]. Available: http://dx.doi.org/10.1007/s11623-017-0761-8.

[8] H. Nissenbaum, *A contextual approach to privacy online*. Stanford, Kalifornien: Stanford University Press, 2011, pp. 32–48.

[9] T. Matzner, P. K. Masur, C. Ochs, *et al.*, “Do-it-yourself data protection – empowerment or burden?”, in *Data Protection on the Move*, Springer, 2016, pp. 277–305.

[10] International Organization for Standardization, *Human-centred design for interactive systems (iso 9241-210:2010)*, Norm, Mar. 2010.

[11] L. A. Liikkanen, H. Kilpiö, L. Svan, *et al.*, “Lean ux: The next generation of user-centered agile development?”, in *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational*, ser. NordiCHI ’14, Helsinki, Finland: ACM, 2014, pp. 1095–1100, ISBN: 978-1-4503-2542-4. DOI: 10.1145/2639189.2670285. [Online]. Available: http://doi.acm.org/10.1145/2639189.2670285.

[12] National Highway Traffic Safety Administration (NHTSA), *Visual-manual NHTSA driver distraction guidelines for in-vehicle electronic devices*, http://www.nhtsa.gov/staticfiles/rulemaking/pdf/Distraction_NPFG-02162012.pdf, Feb. 2012.

[13] B. C. M. Fung, K. Wang, R. Chen, *et al.*, “Privacy-preserving data publishing: A survey of recent developments”, *ACM Computing Surveys (CSUR)*, vol. 42, no. 4, pp. 1–53, 2010.

[14] C. Dwork, “Differential privacy”, in *Automata, Languages and Programming*, ser. LNCS 4052, 2006, pp. 1–12.

[15] D. Chaum and E. Heyst, “Group signatures”, in *Advances in Cryptology - EUROCRYPT ‘91*, ser. Lecture Notes in Computer Science 547, 1991, pp. 257–265.

[16] M. Manulis, N. Fleischhacker, F. Günther, *et al.*, “Group signatures: Authentication with privacy”, Bundesamt für Sicherheit in der Informationstechnik, Tech. Rep., 2012.

[17] G. Ateniese, J. Camenisch, M. Joye, *et al.*, “A practical and provably secure coalition resistant group signature scheme”, in *Advances in Cryptology - CRYPTO ’00*, ser. Lecture Notes in Computer Science 1880, 2000, pp. 255–270.

[18] J. Camenisch and A. Lysyanskaya, “A signature scheme with efficient protocols”, in *Security in communication Networks*, ser. Lecture Notes in Computer Science 2576, 2003, pp. 268–289.

[19] E. Brickell, J. Camenisch, and L. Chen, “Direct anonymous attestation”, in *Proceedings of the 11th ACM Conference on Computer and Communications Security*, ser. CCS ’04, 2004, pp. 132–145.

[20] S. Akl and P. Taylor, “Cryptographic solution to a problem of access control in a hierarchy”, *ACM Transactions on Computer Systems*, vol. 1, no. 3, pp. 239–248, 1983.

[21] S. De Capitani di Vimercati, S. Foresti, S. Jajodia, *et al.*, "Over-encryption: Management of access control evolution on outsourced data", in *VLDB '07 Proceedings of the 33rd international conference on Very large data bases*, ser. VLDB, 2007, pp. 123–134.
[22] I. Corporation, "Intel nuc kit nuc7i5bnk", Tech. Rep., 2017. [Online]. Available: https://ark.intel.com/products/95061/Intel-NUC-Kit-NUC7i5BNK.
[23] Balana Team, *WSO2 Balana Implementation*, 2017. [Online]. Available: https://github.com/wso2/balana.
[24] OASIS Open, *Extensible access control markup language (xacml) version 3.0*, Jan. 2013. [Online]. Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html.
[25] Volkswagen AG, "EXLAP – Extensible Lightweight Asynchronous Protocol (Specification version 1.3)", Tech. Rep., 2012.
[26] D. Crockford, "The javascript object notation (json) data interchange format", Tech. Rep., 2017. [Online]. Available: http://www.json.org/.
[27] S. Duri, M. Gruteser, X. Liu, *et al.*, "Framework for security and privacy in automotive telematics", in *Proceedings of the 2Nd International Workshop on Mobile Commerce*, ser. WMC '02, 2002.
[28] J. Maerien, S. Michiels, S. Van Baelen, *et al.*, "A secure multi-application plattform for vehicle telematics", in *Vehicular Technology Conference Fall*, ser. VTC 2010-Fall, 2010, pp. 1–5.
[29] J. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles", *IEEE Security & Privacy*, vol. 2, no. 3, pp. 49–55, 2004.
[30] *Converge-projekt*. [Online]. Available: http://www.converge-online.de/.
[31] M. Kost, C. Jouvray, M. Sall, *et al.*, "Models and privacy ontology for v2x", in *PRivacy Enabled Capability In Co-Operative Systems and Safety Applications*, Nov. 2010. [Online]. Available: http://cordis.europa.eu/project/rcn/86606_en.html.
[32] *Oversee-projekt*. [Online]. Available: https://www.oversee-project.com/.
[33] BMW Group, *Bmw group startet bmw cardata: Neue und innovative services für den kunden - sicher und transparent*, May 2017. [Online]. Available: https://www.press.bmwgroup.com/deutschland/article/detail/T0271366DE/bmw-group-startet-bmw-cardata:-neue-und-innovative-services-fuer-den-kunden-%E2%80%93-sicher-und-transparent?language=de.
[34] C. Scheiblich and T. Raith, *The Extended Vehicle (ExVe) - New Standardization Project ISO 20078*, Nov. 2014. [Online]. Available: http://taysad.org.tr/uploads/dosyalar/18-12-2014-01-26-5-Extended-Vehicle---a-proposal-for-sharing-diagnostics-data-in-the-future-Scheiblich-ve-Raith-Daimler-27-11-2014.pdf.
[35] "SeDaFa: Selbstdatenschutz im vernetzten Fahrzeug", 2016. [Online]. Available: https://www.sedafa-projekt.de/index.php.