and Values

Jan Grashöfer[1], Alexander Degitz[1] and Oliver Raabe[2]

Abstract: Data we handle and generate when using Internet-based services reveals information on our behavior and values. Access control to this data most often lies in the hands of service or storage providers. With regard to the value ladenness of data sharing in particular, we propose to shift the focus back to the user by introducing the concept of User-Centric Secure Data Sharing (UC-SDS). The UC-SDS approach aims at combining technologies and methodologies from the fields of secure data sharing, secure data outsourcing and value-oriented design. In this paper, we discuss design alternatives using the example of car telemetry data, considering their value interdependencies with particular emphasis on the legal context. Our work indicates that through combination of available building blocks, a higher level of user-centricity in access control is possible than we are used to today.

Keywords: value-oriented design; secure data sharing; secure data outsourcing

1 Introduction

In the light of mobile and cloud computing, social networks and the Internet of Things, we use an increasing amount of Internet-based services every day. Every time we use such services, we give away significant amounts of personal information. Not only the data we submit, but also our access and usage patterns reveal who we are. Data and usage patterns represent our values, thoughts and views. While services, of course, require some user data to provide their functionality, a major research question can be stated as follows: *To which degree can we technically restrict the use of data and observable usage patterns to only those services that are authorized by the user him or herself?*

In this paper, we outline the concept of a User-Centric Secure Data Sharing (UC-SDS) approach. UC-SDS aims at providing the user with a maximum of control over his or her data by reducing necessary trust in third parties. Following the idea of user-centricity, we assess the available technical building blocks with respect to their underlying trust dependencies. We discuss technical means for *Secure Data Sharing (SDS)* and *Secure Data Outsourcing (SDO)*, including methods to hide the patterns of data queries and cryptographically enforced access control, indicating that a much higher level of access control is possible than we are used to today. Furthermore, the methods discussed allow for a separation of data storage from service provisioning and thus significantly contribute to data portability.

[1] Karlsruhe Institute of Technology (KIT), Institute of Telematics, {jan.grashoefer, alexander.degitz}@kit.edu
[2] Karlsruhe Institute of Technology (KIT), Research Group for Information Technology Law and Legal Informatics, raabe@kit.edu

doi:10.18420/in2017_126

However, there is a price tag associated to user-centricity: The downside might be that a user cannot bear the costs or the workload. While a loss of data can cause a loss of reputation and face, a slow service might decrease the service accessibility or the user's satisfaction. Thus, a second research question needs to be addressed: *In which way is User-Centric Secure Data Sharing influenced by values – and how could those be balanced by a user?*

In this paper, we start a discussion on the value ladenness of the UC-SDS approach. Besides individual values, we consider in particular the legal context, reflecting a consensus on societal values. The UC-SDS approach aims at facilitating the economic advantages of modern cloud technologies, without requiring the user to give up data sovereignty.

Our work on UC-SDS was inspired by the concept of user-centric identity management. Shifting the focus towards the user is also a major goal of the *"Solid"*[3] project at MIT led by Tim Berners-Lee. Solid aims at providing "true data ownership" by implementing Web Access Control (WAC)[4]. While the project is in an early stage, protecting stored data using cryptographic mechanisms seems to be out scope for Solid.

Based on a data sharing scenario for telemetry data collected by cars, we will explore the design space of UC-SDS. In recent work, Sookhak et. al. present an approach [So17] that makes use of Proxy Re-Encryption, but the suggested architecture relies on the storage provider to enforce logical access control. Thus, the storage provider is also able to obtain access to the plaintext of data encrypted by the user.

The UC-SDS approach aims at combining technologies and methodologies from the fields of SDS and SDO, considering societal and individual values in the design process as shown in Figure 1.

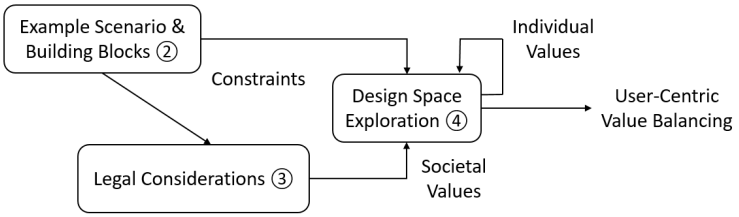

Fig. 1: Structural overview (including section numbers).

2 Example Scenario & Building Blocks

In this section we will sketch a real-world scenario that is used in the course of this paper to motivate, illustrate and discuss the implications of UC-SDS. Furthermore, we will introduce the fields of research that provide the technical building blocks for User-Centric Secure Data Sharing, namely *Secure Data Sharing (SDS)* and *Secure Data Outsourcing (SDO)*, and present related work with respect to *Value-oriented Design*.

[3] https://solid.mit.edu/

[4] https://github.com/solid/web-access-control-spec

2.1 Example Scenario

A frequently considered case yielding the necessity to share data can be found in the field of smart vehicles. A modern car is able to generate extensive amounts of telemetry data per ride that can be used in numerous contexts. Assuming a state-of-the-art electric car, we will consider four actors that are interested in the car's telemetry data:

Manufacturer The manufacturer of the car monitors cars in use to allow target-oriented product improvement. Therefore, the manufacturer is mainly interested in technical data like statistics about motor behavior or battery condition. Additionally, further consumer-oriented services like on-demand maintenance could be implemented based on that data. But, the gathered telemetry data might as well allow competitors to deduce business secrets by in-depth monitoring a car's behavior. Thus the manufacturer is also interested in keeping sensitive subsets of data confidential.

Insurance Company There is in upcoming interest in car telemetry data by insurance companies: Usage Based Insurance (UBI) is a concept that couples insurance rates to usage patterns, rewarding cautious behavior [Ha14]. Consequently the insurance company has a demand for car telemetry data to track the driving profile of their customers.

Power Supplier Another actor interested in car telemetry data might be a power supplier. Approaching smart-grid technologies, controlling and predicting energy demands becomes a key feature of modern power distribution networks [Gu13]. Following the Vehicle-to-Grid (V2G) approach [KT05], electric cars can be used to handle load peaks. In this context information about the usage profile of these cars is of particular value for power suppliers.

User Finally, we refer to the driver of a car as the *user*. The possibility of profiling a user with the help of telemetry data gathered by his or her car causes privacy concerns. Generating all the data discussed before, the user becomes the data subject (c.f. Section 3) and thus, has a fundamental interest in controlling the data, as it implicitly represents his or her individual values. Beginning with location information and driving style up to number of passengers or approximate payload, an extensive amount of private information could be revealed allowing to deduce personal preferences.

The described actors can be interpreted in terms of roles. For example, a car might be used by multiple drivers and usually a plethora of manufacturers is involved in its production. To reduce unnecessary complexity, in the following we will consider the described actors as single subjects. Furthermore, we do not distinguish between driver and owner of a car.

The prevailing approach to gather the necessary data is for each actor to deploy custom-built technologies. Obviously, the preferred way to satisfy all information demands would be to use a standardized built-in system to collect the data. Although technically diverse, the current implementations have in common that they leave the user with little to no control on the data he or she generated. To satisfy the demands of all involved actors, there are essentially two requirements: First, the data has to be stored in a standardized way establishing interoperability. Second, a proper way to share data between the actors has to

be found, while providing the user with a mechanism to control access to the data. The latter aspect is the one we address in this paper.

2.2 Secure Data Sharing

While access control is required for confidentiality and integrity demands, data needs to be shared to allow IT-based collaboration. Conventional data sharing approaches entrust the Storage Provider (SP) to perform logical access control. In case these SPs are considered as *honest but curious* attackers (c.f. Section 4.1), they cannot be trusted to perform access control correctly. In context of Secure Data Sharing (SDS) cryptographic mechanisms are used to enforce access control in untrusted environments, ensuring that only the owner and the members of a sharing group can access the data [KH16, Th14].

In general, SDS offers two basic operations: First, granting access to a resource and second, revoking access to a previously accessible resource. Granting access means to allow decryption, while revoking access corresponds to preventing decryption in the future. Thus, access revocation requires to re-encrypt resources.

A concern with regard to revocation are so called *rollback attacks*, in which a former grantee restores a prior state so that he or she regains access to the resource. Although preventing this attack in general is complex [Li04], the attack vector can be addressed by periodic signatures that indicate freshness in case of only a single writing user.

For the implementation of SDS we consider three different cryptographic mechanisms. Whereas the more advanced cryptographic mechanisms provide additional functionality, they usually come with a performance trade-off [KH16].

Traditional Cryptography Well-established SDS protocols (e.g., [Go03]) make use of common cryptographic primitives like symmetric and asymmetric encryption. These traditional approaches rely on trusted Public Key Infrastructures (PKI) to verify the authenticity of participants and to allow key distribution. All cryptographic operations are carried out on the clients, while the PKI usually represents an external trust anchor.

Proxy Re-Encryption (PRE) Like the traditional approach, PRE [At06] relies on a trusted PKI, but allows to delegate re-encryption of ciphertexts to a proxy that will not be able to access the plaintexts.

Attribute Based Encryption (ABE) While ABE [SW05] relies on a trusted PKI as well, it allows to encrypt data with respect to attributes, which potentially apply to multiple users, resulting in ciphertext that can be decrypted by groups of users. Participants need to trust Attribute Authorities (AA), responsible for attribute binding. Advanced implementations also include PRE for attribute revocation [YJ14].

When talking about SDS, a common implicit assumption is that the data under consideration is stored and eventually accessed multiple times. But given todays high-volume and fast-moving data, processing data streams directly has become a considerable alternative. Stepping back from the afore mentioned assumption, the Data Stream Processing paradigm represents another domain of SDS. In this context, enforcing access control for streamed data is a hot topic of current research [Th16].

2.3 Secure Data Outsourcing

Secure data outsourcing has been an active research topic for the last two decades. The underlying question is: *How can encrypted data be queried efficiently?* An early approach which addressed this problem is called Searchable Encryption (SE) [Ka12]. This approach has the data encrypted and still has fast access times on data, but it comes with two drawbacks: First, this approach only works with a deterministic encryption scheme, which is vulnerable to inference attacks as described in [Na15]. Since the same keyword is encrypted to the same ciphertext, an attacker can analyze the frequency of equal ciphertexts and compare the frequency to public data, to infer plaintexts. Second, SE approaches leak access patterns of data. Similar to the description above, a more powerful attacker, who has access to the data and the network traffic to and from the SP, can also analyze the frequency of queries and responses of a certain data set and perform an inference attack on this information.

These two drawbacks combined lead to more recent approaches, which use an encrypted index structure that is able to work with probabilistic encryption schemes and that obfuscates access patterns with mechanisms like data node shuffling and the retrieval of dummy elements [St13]. This is done by a mediator server, which all database queries are directed to. In these approaches, the mediator is the only entity that directly communicates with the database. The mediator has to be hosted in a trusted environment to ensure that the access patterns of queries are hidden. While these mechanisms add an overhead in access times and network traffic, they also provide a much higher level of confidentiality.

2.4 Value-oriented Design

To provide a value-oriented service, the consideration of values already has to start in the design phase. It is difficult to express values like power, self-direction and benevolence in well-defined functional or non-functional requirements during the design phase. An additional challenge is that almost all software products carry value conflicts between different stakeholders.

One approach to bridge the gap between values and system requirements or technical properties was proposed by Van de Poel [VdP13]. Based on value-sensitive design [Fr96], he suggests an intermediate layer between values and system requirements, which is called norms. Norms instantiate values in form of individual or societal restrictions to a certain scenario. In this paper we distinguish between societal values, e.g. inherent to the law (c.f. Section 3), and individual values like “self-direction” or “benevolence” (c.f. Section 4.4). By analyzing the legal framework, we consider societal values stated in the law. In addition, we use a top-down approach to exemplify how individual values (based on the refined value model of Schwartz [Sc12]) affect our proposed concept. Similarly to [Ne16], we outline which normative elements can be expressed through which design alternatives for an example scenario. By making explicit how the proposed normative elements affect the design decision, we can visualize possible value conflicts.

3 Legal Considerations

The societal values to be considered with regard to UC-SDS are related to the question by which legal means a *"European data economy"* could be developed. The automotive sector could serve as a blueprint, as regulatory efforts are already advanced. In this paper we use the German law as a starting point to consider national and European legislation. The most urgent preliminary question is *"who should have the right to process and monetize vehicle-related data"* [Ho15, p. 359]. Recent efforts[5] show that the European Commission evaluates the possibilities for addressing the issue of access to sensor data.

3.1 Economic Perspective

From an economic perspective the Commission discusses the justification of a legal framework for a **"Data Ownership"[6] versus an "Access Right"** based on competition law. Concerning **"Data Ownership"** it is mostly assumed that due to the absence of physicality there is no ownership or property right of data under civil law [BJ16, p. 413]. Nevertheless, some legal scholars assume a similar right for the vehicle sector with regard to Section 303a German Criminal Code (StGB) [Vo16, p. 6]. This assumption of ownership is questionable. If data could be owned, the owner would have exclusive control over the data. However, vehicle-related data is not only of technical nature, but also contains information about the person using a vehicle [Ku16, p. 510]. While Buchner argued that vehicle-related data would be "ephemeral and therefore not relevant for data protection law" [Bu15, p. 374], the argument does not hold for modern big data applications. Hence, privacy has to be considered in terms of *data protection law* (see Section 3.2). This should apply, even if a driver cannot be identified, to avoid restraints upfront [Ho15, p. 361]. Taking into account that data ownership would be transferable, while the data subject remains the same, one can argue against a *"Data Ownership"* that privacy is a fundamental right and, ultimately, part of the human dignity (Art. 1 para. 1, Art. 2 para. 1 German Constitution). From a legal point of view it is not possible to trade-in the right to informational self-determination or sell exclusive rights concerning personal data. This would involve a possibility to waive the right of human dignity, which is not foreseen under the German Constitution. As data subjects have the right of erasure, "Data Ownership" would also be constantly limited and subject to a pending condition of revocation. Additionally the distinction between personal and non-personal data is not always clear and more often even fluid over time. Overall, the concept of "Data Ownership" is inappropriate, in order to achieve legal certainty in the data-driven economy.

This conclusion does not answer the question whether and how a legal framework concerning an **"Access Right"** based on competition law could be established. The European General Data Protection Regulation (GDPR) encompasses the protection of privacy as well as the *"free flow of data"* as regulatory targets. The Commission remarks that "privacy concerns

[5] Commission Staff Working Document on the free flow of data and emerging issues of the European data economy. Accompanying Communication Building a European data economy COM(2017) 9 final.

[6] The "Ownership"-metaphor can be misleading: Here, the term is used in the legal sense of physical possession.

are legitimate concerns but should not be used as a reason to restrict the free flow of data in an unjustified way. The GDPR fosters the necessary trust for data processing and is the foundation for the free flow of personal data in the EU"[7]. Considering this background, another relevant topic is the *"right to data portability"* in Art. 20 GDPR. According to the common understanding, this right was established to pursue competitive purposes in order to avoid "lock-in-effects", corresponding to antitrust law, or a monopolistic collection of data [Pa17, on Art.20 Rn.6]. As Competition Law already regulates market-dominant ventures, the remaining question is: How to adapt the existing competition law framework to modern data sharing technologies.

During the consultation process started by the EU-Commission[5], access to and re-use of personal data with respect to data generated in cars are part of the legal discussion. The issue of access to in-vehicle data has been discussed in the framework of a Cooperative Intelligent Transport Systems (C-ITS) stakeholder platform[8]. Primarily, a set of guiding principles for access to in-vehicle data have been developed, including: **"fair and undistorted competition"**, prior **"consent"** of the data subject, **"standardized access"** and **"free choice"** of the data subject. Moreover relevant to our approach it is also recommended to foster the principle of *"Privacy by Design"*, as it has been requested in German literature [KW15, p. 387]. At the moment discussions focus on a data server platform in an *"Extended Vehicle Concept"*, while the control of the access conditions and extent of the in-vehicle data is in the hands of the manufacturers. An alternative concept proposed another implementation of the data server platform, managed and controlled by a neutral third-party. By incorporating the established principles in the design of UC-SDS, we aim at providing another option.

3.2 Data Protection Perspective

The main requirements of data protection law which must be met by UC-SDS should be based on the GDPR's key principles. With regard to access to data the GDPR takes – as already set out above – also aspects of economic competition into account. Hence, interpretation of the protective provisions of the GDPR in the context of technical access control must also take aspects of competition as well as data protection into due consideration.

Therefore, the interpretation of the relevant key principles – especially the principle of **"Data Minimization"** as set out in Art. 5 GDPR and the requirement of **"Privacy by Design"** as laid out in Art. 25 GDPR – must aim at reciprocal optimization of competition as well as data protection. A technology design based on the principle of data minimization has been interpreted in literature as a right to "data-free mobility" [Ku16, p. 510]. This would mean, that from a data protection point of view only small amounts of data could be transmitted to the data serving platform and transmission must be done under consideration of possible means of pseudonymization. In the past, methods of data minimization for connected cars, like frequent change of pseudonyms, did not function sufficiently well [Ha15, p. 371]. While

[7] Communication from the Commission to the European Committee of Regions, "building an European data economy", COM(2017) 9 final, Brussels, 10.1.2017, p. 5.

[8] See `http://ec.europa.eu/transport/themes/its/c-its_en`.

the wording of Art. 5 GDPR indicates the necessity to prevent the collection of personal data, a way of data minimization could also be seen in the possibilities of encryption. By encrypting data and providing cryptographic access control to the consumer, privacy risks may be minimized, while competitors' interests in using this data may be balanced out. Additional problems discussed in literature regarding *"responsibility"* for compliance with data protection regulations [Ro16, p. 111 f.] are far easier to solve as well. Since the platform operator himself does not have access to the data, responsibility to fulfill data protection obligations falls to the service providers accessing the data.

When it comes to the technical design, the most serious problems encountered refer to **"transparency"** and **"consent"**, which is necessary to legitimate the data processing in most cases. Some experts consider the actual regulations concerning data protection information as insufficient to provide the person, to whom the car is registered to, and the driver with the required *"transparency"* about current collection and processing of personal data [Ro16, p. 111 f.]. Even if this requirement is narrowed down to the knowledge of facts that are crucial for the data subjects decision, whether or not to give consent for processing of personal data, it is still difficult to guarantee sufficient information as required in Art. 7 GDPR [Bu15, p. 372].

3.3 Access Control Approach

We showed that the regulatory approach of an "Access Right", in contrast to the idea of "Data Ownership", could in principle provide a better balance between economic regulation and data protection from a legal perspective. Furthermore, we outlined that, in the light of an upcoming "European data economy", the classical Data Minimization paradigm is contradictory to the concepts of Big Data. With respect to the potential of Big Data applications, we suggest shifting from Data Minimization towards user-enforced cryptographic access control, as a possible compromise. Following this approach, competition as well as data protection concerns could be brought together in a well-balanced manner. In the following, we will incorporate the previously described legal key principles, representing a societal canon of values, into the design of UC-SDS by considering them in the process of establishing technical requirements.

4 UC-SDS Design Space Exploration

In this section we specify the UC-SDS approach by outlining its foundations. Then, we analyze the design space of UC-SDS with respect to the technical building blocks introduced in Section 2. Finally, we provide an example on how design decisions can be supported by considering individual values.

4.1 Foundations

Legal Framework As shown in Section 3, from an economic perspective, UC-SDS has to guarantee *fair and undistorted competition*, avoiding data access by an omniscient

market role (e.g., a storage provider). Hence, data should be encrypted. Due to his or her fundamental rights regarding *data privacy*, the consumer should be in the sole position to decide who should have access to the vehicle-data by performing cryptographic access control. Ensuring this could also avoid a targeted hindering of competitors (Section 4 Nr. 10 UWG) or a violation of antitrust law [Ho15, p. 365]. Finally, enforcing interoperability also allows the consumers' *free choice* to select services.

Additionally, the UC-SDS approach is able to bring together competition and data protection concerns. With regard to the mandatory withdrawal of *consent*, cryptographic access control has an advantage compared to conventional methods (e.g., [Bu15, p. 377]), because withdrawing consent can be realized straightforward by revoking the access permissions. Since withdrawal of consent concerns only future processing of data, the legal principles are duly taken into account. Another requirement that can be deduced for an UC-SDS implementation is that giving consent can be done separately for each service. Furthermore, the user must be able to separately withdrawn consent in a similar way [Ku16, p. 514].

Data Model In context of SDS, the data model determines the required granularity of the SDS mechanism to be applied. The *granularity* describes the level of access control enforcement, i.e. what is considered as the smallest entity to be controlled. Thus, we distinguish three elementary forms of data: First, *data blobs* with arbitrary internal structure, e.g., files, representing the most common use case for SDS. In case of files, external structuring like directories has to be considered as well. Second, *databases* containing multiple records of structured data. In case of databases we assume a fixed scheme providing the basis for access control policies. Finally, data can be provided in form of continuous *data streams* that are not necessarily stored but processed on the fly.

In this paper we will discuss UC-SDS based on the scenario described in Section 2.1. Given the structure of telemetry data, the generated data might be processed following the data stream paradigm or stored and queried using a database. Considering varying actors, e.g., due to switching the insurance company or power supplier, we choose to support access to historical data by using a database as an example.

Attack Model In this paper we consider an *honest but curious attacker*, who has access to the network connections between all participants and to the physical data, as it is stored on the server of one of the participants. This includes the actors or the Storage Provider themselves as possible attackers.

Requirements The following *security requirements* must be fulfilled by any solution to be considered for the UC-SDS approach: **(S1)** An attacker can never see the plain text of outsourced data. **(S2)** At any given time, actors can only access the data they are allowed to access. **(S3)** Attackers should infer as little as possible through access patterns.

In addition, the following *functionality requirements* must be met: **(F1)** A user must stay in control of his or her shared and outsourced data. **(F2)** It must be visible which data is shared with whom. **(F3)** Access rights to data must be cryptographically granted and revoked. **(F4)** Data must be accessible to actors, independent of the users availability.

4.2 Secure Data Sharing Aspects

When it comes to Secure Data Sharing the basic question to answer is: *Whom do we need to trust?* The core of a Secure Data Sharing approach is its protocol, which defines how access and key management interactions are carried out. The protocol codifies different roles and their relations based on the underlying cryptographic primitives. Although the resulting architectural designs may vary for different protocols, the underlying trust anchors are determined by the cryptographic mechanisms in place. It is up to the designer, where to cast the anchors by composing the protocol. Therefore, we will assess the feasibility of different SDS approaches by considering the trust models inherent to the available cryptographic primitives:

Traditional Cryptography Using traditional cryptographic mechanisms we can fulfill the first two security requirements defined in Section 4.1: By encrypting the users' data we ensure that an attacker cannot access the plaintext (S1). By sharing the encryption key, we can grant access to a chosen subset of actors (S2). To revoke access, the user needs to re-encrypt the data and share the new key. In context of the telemetry scenario, the car itself is able to detect rollback attacks by monitoring the freshness of the data. The functional requirements (F1) and (F2) need to be addressed by providing a sufficient management interface to the user. (F4) requires the actors to be able to access the outsourced data regardless of the user's availability. This can be realized by using encrypted keys, called *Lockboxes* [KH15], which are stored together with the data. The main drawback of this concept is the fact that the user's device is responsible for re-encryption. In case of large amounts of telemetry data, this could be a time-consuming task.

Proxy Re-Encryption (PRE) To increase the flexibility in case of re-encryption, we consider two different approaches: First, the user explicitly determines the time of re-encryption and provides the necessary resources. Second, re-encryption could be outsourced. Using PRE, the proxy would not be able to access the plaintext, satisfying security requirement (S1). In contrast, the security requirement (S2) as well as the functional requirements (F1) and (F2) might be violated: A malicious proxy, could suppress re-encryption requests of the user. Consequently, the user has to put some level of trust into the re-encryption proxy. With respect to the scenario described in Section 2.1, two options for mapping the proxy-role to participants seem reasonable: **(P1)** Trusting the *service provider* to execute re-encryption. **(P2)** Trusting the *manufacturer* to execute re-encryption. As the SP is assumed to be honest but curious, (P1) complies with the attacker model in place. Only in case the SP cooperates with malicious actors, using PRE would open an attack vector. Hence, outsourcing re-encryption to the manufacturer (P2) seems counterproductive: As the manufacturer is one of the considered actors, taking the role of the re-encryption proxy at the same time would allow him to retain access on data. But, given that the manufacturer is in charge of the car's hardware, the user already has to trust the manufacturer. Thus, the second option would effectively reduce the necessary trust relations.

Attribute Based Encryption (ABE) The idea of user-centricity is to provide the user with a maximum of control over his data. Thus, the user will be imposed with the full

responsibility to judge the trustworthiness of all involved actors. In a real world scenario, this assumption is rather unrealistic. ABE can be used to reduce complexity by implementing levels of indirection. Introducing trusted AAs, the user is still able to enforce custom access policies, while trust relations are simplified. For example, a national regulatory authority, acting as an AA, could bind a dedicated attribute to insurance companies that comply with data protection laws.

SDS increases computational costs due to the additional cryptographic operations. Furthermore, advanced cryptographic primitives typically cause higher costs than traditional approaches. But, due to the nature of user-centricity, any UC-SDS system is limited in terms of complexity to what is manageable for a single user. With respect to recent performance evaluations [KH15, KH16], we conclude that performance considerations in terms of trade-offs between different cryptographic methods are negligible regarding an SDS protocol applied in context of UC-SDS.

4.3 Secure Data Outsourcing Aspects

Secure Data Outsourcing mechanisms enrich our concept to meet two requirements, namely indexing for fast data access and low information leakage to protect the data from inference attacks. The question to answer is: *How can we store and retrieve data securely and efficiently?* There is an inherent trade-off between retrieving data securely and efficiently. More efficient indexes, even encrypted ones, leak more information about the data, because a potentially untrusted actor gains information which is needed to perform queries more efficiently. Our goal is to only leak information which the attacker either already knows or with which he can not infer sensitive data.

Fast Index Access (FIA) We assume that each data set contains a summary of the telemetric driving data for one day and that one data set is uploaded per day. In this case all actors who want access to the data already know that a data set is inserted once every day, so there is no point in trying to hide or mask this insert. The date field of the data set makes a good index, as it is known to all participants and thus, there is no need to encrypt it. All other columns should be encrypted with a probabilistic encryption scheme. Otherwise, especially the manufacturer, as a possible attacker who has a vast amount of background knowledge, might infer big parts of data, encrypted with a deterministic or order-preserving encryption scheme, by performing a frequency analysis.

With this setup, it is possible for each actor to query data sets according to their date. The whole data set is retrieved even if an actor only has access to one value, so there is a considerable amount of overhead in network traffic. An attacker as defined in Section 4.1 can see for every query (1) who retrieves data, and (2) how many and (3) which data sets, identified by their date, are retrieved. Additionally, an attacker can keep track on (4) how often a certain data set has been retrieved.

Low Information Leakage (LIL) To use the security mechanisms of an encrypted index structure, we need to have a mediator server to execute shuffling operations or to retrieve

dummy data sets to hide access patterns. There are several possibilities to place such a mediator: **(M1)** A user could operate his own server, which works as a mediator, **(M2)** The mediator software could be run on the server of the storage provider, **(M3)** The mediator software could be run on a server of the car manufacturer, making him the central database access point for all participants, **(M4)** The mediator software could be installed at every party's server, which wants to access data.

Even though the options (M1) and (M4) are user-centric, truly they do not seem practical for many scenarios, since they involve a lot of manual installation and maintenance on either the user's side or the side of all involved actors. If they can be implemented, they are definitely to be preferred. Whether option (M2) or (M3) is a better choice highly depends on the scenario. In our case we would opt for (M3), since the manufacturer knows some of the information we would hide with an encrypted index structure anyways, because he knows the kind of data that is produced by the car. The car manufacturer can outsource encrypted data on behalf of the user with a protocol like Oblivious RAM. This means that every database operation would be send to the car manufacturer, and then executed by him. The key management and the encryption of data, and therefore the management of access control would remain with the user. This solution eliminates the information leakage of (1), (3) and (4) as defined above while still remain a decent query latency.

4.4 Design and Value Considerations

To implement a UC-SDS system, we can combine the previously discussed building blocks of SDS and SDO. Figure 2 shows how a solution for low information leakage might be combined with proxy re-encryption. In this example, the mediator as well as the re-encryption proxy, which acts like another user from the mediator's perspective, are operated by the manufacturer. Thus, the manufacturer is entrusted with hiding access patterns and the timely execution of re-encryption, in case a user revokes access of an actor. As the sketched design is only one of many possibilities to combine the available building blocks, an additional questions comes up: *How can we decide on the various design alternatives?*

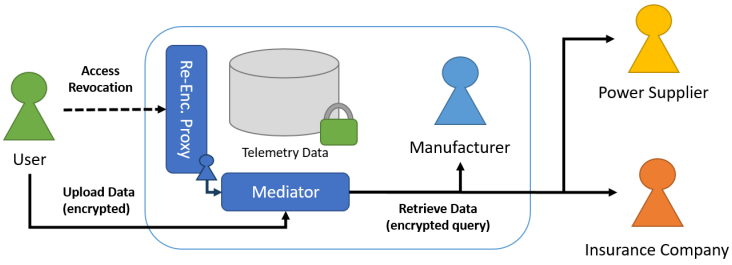

Fig. 2: UC-SDS design combining LIL (M3) and PRE (P2).

At this point we suggest to incorporate individual values, i.e. values of the user, following the approach of [Ne16], in which values are translated to technical properties. This approach is being used to grant a starting point for a discussion about the implications on values

of each solution. We have identified that, among others, the following values affect our UC-SDS approach: **Power - Resources** as "Power through control of material and social resources", which is affected by the monetary cost of any given solution. **Self-Direction - Action** as "Freedom to determine one's own actions", which is affected by the number of operation modes a user can choose from. **Benevolence - Dependability** as "Being a reliable and trustworthy member of the ingroup", which is affected by the amount of trust towards others, that is necessary for a chosen solution. From these values we can derive four normative elements, which will help to evaluate the three values with regard to our scenario: *"Minimize Cost"* expresses the requirement to keep operating costs as low as possible. *"Individual Protection Levels"* expresses the requirement to allow differing protection levels for different data attributes. For example, uncritical data attributes can be stored unencrypted to boost performance. *"Large Design Space"* expresses the requirement to set up a very individual solution with many operation modes to choose from. *"Minimize necessary trust"* expresses the requirement to minimize the trust in third parties, which is needed to operate the chosen solution. Figure 3 aids a discussion about design alternatives

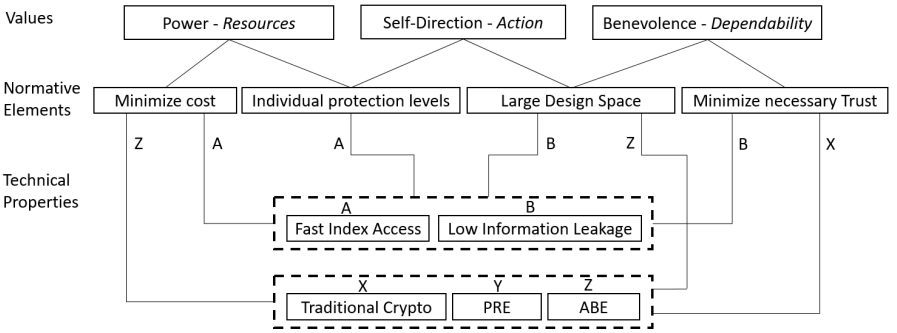

Fig. 3: Dependencies of Values, Normative Elements and Technical Properties. Annotations indicate preferred design decisions with respect to a single Normative Element.

on a very basic and simplified level in taking affected values into account. The letters on relations express, which design option fulfills a normative element best. For example, it can be seen that A (Fast Index Access) minimizes cost better than B (Low Information Leakage) or that X (Traditional Crypto) minimizes necessary trust better than Y (PRE) and Z (ABE). The visualization also shows the implicit trade-offs between values. Option A is better in terms of minimizing cost and providing individual protection levels of data, whereas option B provides a large design space.

5 Discussion

We have presented an **example scenario** dealing with telemetry data collected by cars to illustrate the implications of UC-SDS. While this scenario clearly shows the potential of UC-SDS, constraining assumptions had to be made with respect to the scope of this paper. For a real-world application it has to be considered that the driver of a car might not

be the owner and might change frequently. Likewise, the role of the manufacturer has to be carefully reviewed. On the one hand, a car manufacturer is in charge of implementing the data collection mechanisms, implicitly controlling which data is acquired and thus accessible for others. It has to be guaranteed that a manufacturer cannot exploit this central position. Assuming sufficient competitive pressure this might be a self-regulating process. On the other hand, even in case of UC-SDS there are legitimate data-processing-operations performed by the manufacturer that should not be prohibited by the user. This concerns data about security-relevant components. Manufacturers have to fulfill their product monitoring obligations to prevent liability risks. A UC-SDS system has to be designed in a way that access to those data is ensured for manufacturers. In context of the scenario, the selected **data model**, described in Section 4.1, is up to discussion as well. Considering the actual information needs of the involved actors, a more fine-grained specification will be required.

From the **legal perspective**, we showed that UC-SDS enables to balance the main legal principles on competition- and data protection law level: For example, UC-SDS fosters "fair and undistorted competition" by providing confidentiality and realizes "consent" by enabling the user to perform access control. According to the value model of Schwartz [Sc12], the European regulation efforts can be understand as an approach to establish a framework to balance individual values. In this context, the interrelations of societal and individual values should be considered for further discussion, e.g., with respect to liberty rights. The **practical usability** of UC-SDS heavily depends on the possibilities of the user to interact with the system. For example, given the telemetry data scenario, a user interface could be part of the car or a separate (mobile) application. The underlying question, which level of complexity a user can handle, is beyond the scope of this paper and remains to be addressed.

Overall, we have outlined that the necessary technical building blocks to implement UC-SDS are available. More work is needed to develop a protocol and architecture with respect to the particular use case. To decide on the various design alternatives, we suggest taking **values** into account. While societal values determine requirements for UC-SDS, e.g. in form of laws, we sketched how individual values and value-conflicts can be considered as well. Centering on the user, a preeminent trade-off manifests between minimizing trust relations ("benevolence") and cost minimization ("power"), as cutting down trust relationships significantly increases the users' workload. Given the complexity of the discussed scenario, the consideration of values has to be intensified. Finally, following a user-centric approach itself represents a value-laden decision.

6 Conclusion & Future Work

In this paper we have introduced the idea of User-Centric Secure Data Sharing to restrict the use of data and observable usage patterns to services authorized by the user. Based on a data sharing example scenario for telemetry data collected by cars, we have analyzed the legal context, representing a consensus on societal values to deduce scenario specific requirements. With respect to the established requirements, we have analyzed the available technical building blocks. We exemplified how potentially conflicting individual values

can be considered when combining these building blocks to balance the various trade-offs inherent to the possible design alternatives.

Overall, we come to the conclusion that the technical means to implement UC-SDS are available, while possible combinations of the respective building blocks have not been thoroughly investigated yet. In addition to improving usability, UC-SDS requires a careful analysis of societal and individual values, according to the scenario under consideration, to realize a well-balanced solution. Thus, our work represents the first step towards a value-oriented design of User-Centric Secure Data Sharing.

For future work, we plan to thoroughly examine a UC-SDS scenario, identifying a comprehensive set of affected values. Implementing the building blocks, we envisage a prototypical realization, which can be used to investigate the practical feasibility of UC-SDS.

References

[At06] Ateniese, Giuseppe et al.: Improved proxy re-encryption schemes with applications to secure distributed storage. ACM TISSEC, 9(1):1–30, February 2006.

[BJ16] Brisch, Klaus; Jung, Marco Müller-ter: Autonomous Driving – Von Data Ownership über Blackbox bis zum Beweisrecht. Computer und Recht, 32(6), January 2016.

[Bu15] Buchner, Benedikt: Datenschutz im vernetzten Automobil. Datenschutz und Datensicherheit - DuD, 39(6):372–377, May 2015.

[Fr96] Friedman, Batya: Value-sensitive design. interactions, 3(6):16–23, December 1996.

[Go03] Goh, Eu-Jin et al.: SiRiUS: Securing Remote Untrusted Storage. In: NDSS. volume 3, pp. 131–145, 2003.

[Gu13] Gungor, Cagri et al.: A Survey on Smart Grid Potential Applications and Communication Requirements. IEEE Transactions on Industrial Informatics, 9(1):28–42, February 2013.

[Ha14] Handel, Peter et al.: Smartphone-Based Measurement Systems for Road Vehicle Traffic Monitoring and Usage-Based Insurance. IEEE Systems Journal, 8(4):1238–1248, December 2014.

[Ha15] Hansen, Marit: Das Netz im Auto & das Auto im Netz: Herausforderungen für eine datenschutzgerechte Gestaltung vernetzter Fahrzeuge. Datenschutz und Datensicherheit - DuD, 39(6):367–371, May 2015.

[Ho15] Hornung, Gerrit: Verfügungsrechte an fahrzeugbezogenen Daten: Das vernetzte Automobil zwischen innovativer Wertschöpfung und Persönlichkeitsschutz. Datenschutz und Datensicherheit - DuD, 39(6):359–366, May 2015.

[Ka12] Kamara, Seny et al.: Dynamic Searchable Symmetric Encryption. In: ACM CCS. New York, NY, USA, pp. 965–976, 2012.

[KH15] Kuehner, Holger; Hartenstein, Hannes: On the Resource Consumption of Secure Data Sharing. In: IEEE RATSP. pp. 880–889, August 2015.

[KH16] Kuehner, Holger; Hartenstein, Hannes: Decentralized Secure Data Sharing with Attribute-Based Encryption: A Resource Consumption Analysis. In: ACM SCC. pp. 74–81, 2016.

[KT05] Kempton, Willett; Tomić, Jasna: Vehicle-to-grid power fundamentals: Calculating capacity and net revenue. Journal of Power Sources, 144(1):268 – 279, 2005.

[Ku16] Kunnert, Gerhard: Die datenschutzkonforme Vernetzung des Automobils. Computer und Recht, 32(8):509–516, January 2016.

[KW15] Krauß, Christoph; Waidner, Michael: IT-Sicherheit und Datenschutz im vernetzten Fahrzeug: Bedrohungen und Herausforderungen. Datenschutz und Datensicherheit - DuD, 39(6):383–387, May 2015.

[Li04] Li, Jinyuan et al.: Secure Untrusted Data Repository (SUNDR). In: OSDI. volume 4, pp. 9–9, 2004.

[Na15] Naveed, Muhammad et al.: Inference attacks on property-preserving encrypted databases. In: ACM SIGSAC CCS. pp. 644–655, 2015.

[Ne16] Neudecker, Till et al.: Consideration of Values in the Design of Access Control Systems. In: Proceedings of a Workshop on Value-Sensitive Design of Internet-Based Services: Towards an Integration of Technology and Values. GI'16, Klagenfurt, 2016.

[Pa17] Paal, Boris P. et al.: Datenschutz-Grundverordnung. Beck'sche Kompakt-Kommentare. C.H. Beck, München, 2017.

[Ro16] Roßnagel, Alexander et al.: Datenschutzrecht 2016 „Smart" genug für die Zukunft? kassel university press, Germany, 2016.

[Sc12] Schwartz, Shalom H. et al.: Refining the theory of basic individual values. Journal of Personality and Social Psychology, 103(4):663–688, 2012.

[So17] Sookhak, Mehdi et al.: Secure Data Sharing for Vehicular Ad-hoc Networks Using Cloud Computing. In: Ad Hoc Networks, volume 184, pp. 306–315. Springer International Publishing, Cham, 2017.

[St13] Stefanov, Emil et al.: Path ORAM: An Extremely Simple Oblivious RAM Protocol. In: ACM SIGSAC CCS. pp. 299–310, 2013.

[SW05] Sahai, Amit; Waters, Brent: Fuzzy Identity-Based Encryption. In: Advances in Cryptology – EUROCRYPT 2005, volume 3494, pp. 457–473. Springer Berlin Heidelberg, 2005.

[Th14] Thilakanathan, Danan et al.: Secure Data Sharing in the Cloud. In: Security, Privacy and Trust in Cloud Systems, pp. 45–72. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.

[Th16] Thoma, Cory et al.: PolyStream: Cryptographically Enforced Access Controls for Outsourced Data Stream Processing. In: ACM SACMAT. pp. 227–238, 2016.

[VdP13] Van de Poel, Ibo: Translating Values into Design Requirements. In: Philosophy and Engineering: Reflections on Practice, Principles and Process, volume 15, pp. 253–266. Springer Netherlands, Dordrecht, 2013.

[Vo16] Vogelgesang, Stephanie: Datenspeicherung in modernen Fahrzeugen – wem „gehören" die im Fahrzeug gespeicherten Daten? juris – Die Monatszeitschrift, 3(1):2–8, 2016.

[YJ14] Yang, Kan; Jia, Xiaohua: DAC-MACS: Effective Data Access Control for Multi-Authority Cloud Storage Systems. In: Security for Cloud Storage Systems, pp. 59–83. Springer New York, 2014.