Institut für Parallele und Verteilte Systeme

Universität Stuttgart
Universitätsstraße 38
D–70569 Stuttgart

Bachelorarbeit

Repository zur Verbesserung des Datenschutzes für vernetzte Fahrzeuge

Annika Egeler

Studiengang:	Informatik
Prüfer/in:	Prof. Dr.-Ing. habil. Bernhard Mitschang
Betreuer/in:	Yunxuan Li M.Sc, Dr. re. nat. Christoph Stach
Beginn am:	2. Mai 2023
Beendet am:	2. November 2023

Kurzfassung

Durch das bleibende Interesse und die steigende Anzahl von Features in Fahrzeugen steigt die Konnektivität zwischen den Fahrzeugen und zwischen Fahrzeug und Cloud. Dadurch steigen die Bedenken der Nutzer in Bezug auf ihre Privatsphäre. Viele fürchten, dass ihre privaten oder auch intimen Daten ungefiltert gespeichert, verarbeitet und weitergegeben werden. Dadurch könnte es möglich sein, das Fahrzeug zu echten Personen, ihrem Wohnort oder persönlichen Details zuzuordnen und diese zu Werbezwecken oder anderen Zwecken zu verwenden. Daher ist es wichtig, auf diese Bedenken einzugehen und Lösungen für den Schutz der Privatsphäre zu entwickeln. Dazu wurden bereits einige Techniken entwickelt. Diese sind jedoch für Endnutzer oft schwer zu finden, da sie nicht an einem Ort gesammelt sind. Außerdem benötigen sie Expertise, um sie zu verstehen. Das liegt daran, dass keine Übersicht erstellt wurde, die verständlich erklärt, was diese Techniken können und wie sie eingesetzt werden. Auch gibt es nicht eine Technik, die für jeden Nutzer passend ist, da diese ihre Privatsphäre unterschiedlich stark schützen wollen oder ihnen bei unterschiedlichen Daten wichtig ist diese zu schützen. Dadurch ist es für den Nutzer schwer, eine für ihn passende Technik zu finden. Um die Techniken also für Endnutzer zugänglich zu machen, ist es nötig, eine solche übersichtliche Sammlung zu erstellen.

In dieser Arbeit wird ein selbst erstelltes System, mit dem Namen PriTeX vorgestellt, in dem einige dieser Techniken enthalten sind. Dort soll es möglich sein, die Techniken zu speichern, zu durchsuchen und zugehörige Informationen zu ihnen zu finden. Um das finden und durchsuchen leichter zu machen, soll ein Metadatenmodell entworfen werden. Dieses soll die wichtigsten Eigenschaften der Techniken abbilden und diese dadurch in sinnvolle Kategorien einteilen. So soll es einem Nutzer möglich werden, eine für seine Anforderungen und Wünsche passende Technik zu finden und so seine Privatsphäre zu schützen. Das erstellte System besteht aus dem Backend, einem Repository, welches die Techniken und Metadaten speichert und einer Webseite, welche eine graphische Oberfläche für das System darstellt.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Verzeichnis der Listings

Abkürzungsverzeichnis

CV Connected Vehicle. 15

DoS Denial of Service. 43

IoT Internet of Things. 68

LBS Location-based Service. 38

MPC Multi-party Computation. 40

PET Privacy Enhancing Technologie. 16

PJD Protection Jamming Device. 42

PriTeX **Pri**vacy **T**echnologie **Ex**change. 16

PVA Personal Voice-Assistant. 16

SpAn Speed Anonymization. 37

VCDA Voice-controlled Digital Assistant. 42

1 Einleitung

Die aktuelle Entwicklung der Technik geht hin zu immer mehr Funktionen, die das Leben leichter und sicherer machen sollen. Um das zu erreichen, werden immer mehr Sensoren eingesetzt. Diese werden verwendet, um die neuen Funktionen zu implementieren und die Sicherheit zu gewährleisten. Durch die steigende Anzahl an Sensoren im Umfeld von Privatpersonen entsteht jedoch ein Problem für deren Privatsphäre. Einige Möglichkeiten, aber auch Bedrohungen, die sich durch die aufgezeichneten Daten ergeben, werden von Stach et al. [SGB+22] beschrieben. Durch diese Bedrohungen verlangen viele einen Schutz der Privatsphäre. Dieser soll guten Schutz bieten, aber dennoch einfach nutzbar sein.

Gerade die Entwicklungen in der Fahrzeugindustrie schaffen in dieser Hinsicht Komplexität. Der Einsatz von vielen Assistenzsystemen, wie Spurhalteassistenten, Parkassistenten, Tempomat mit Abstandshalter und vielen weiteren, erfordert den Einsatz von Sensoren, wie Kameras oder Geschwindigkeitsmessern. Wie von Yang et al. [YJZ+18] beschrieben, können diese Sensoren in das Auto oder nach außen gerichtet sein. Nach innen gerichtete Sensoren sind beispielsweise Reifendrucksensoren, Motortemperatursensoren und viele weitere. Die nach außen gerichteten Sensoren können Kameras oder verschiedene Radarsensoren sein. Dabei sind die Sensoren allein jedoch nicht das Problem. Um die Funktionen und Sicherheit eines Fahrzeuges zu verbessern, ist dieses mit weiteren Parteien, wie dem Hersteller verbunden und tauscht Daten aus. Fahrzeuge, die sich das zunutze machen, werden von Uhlemann [Uhl15] auch als vernetzte Fahrzeuge oder Connected Vehicles (CVs) bezeichnet. Der Datenaustausch findet statt, da manche Funktionen nicht lokal umsetzbar sind oder um Analysen über das Fahrzeug durchzuführen. Die gesammelten Daten werden oft ungefiltert weitergegeben und an einem anderen Ort verarbeitet. Wieler et al. [WKH23] haben einige Daten gefunden, die von den Fahrzeugen ungefiltert weitergegeben werden. So werden beispielsweise GPS-Positionen oder Ladung der Antriebsbatterie weitergegeben. Daher ist es wichtig, dass der Nutzer eines CVs selbst Möglichkeiten zur Filterung der Daten zur Verfügung hat, um so seine Privatsphäre schützen zu können.

Was ohne den Schutz der Privatsphäre passieren kann, wird außerdem klar, wenn man auf den Artikel von Reuter schaut, in welchem beschrieben wird, wie sensible Informationen an Mitarbeiter weitergegeben wurden [SCJ23]. In dem Artikel geht es um die Fahrzeuge von Tesla, die selbstfahrend und vernetzt sind. Sie zeichnen unter anderem Kamera- und Stimmdaten auf und das auch in privaten Bereichen wie dem Garten oder der Garage der Besitzer. Es wird erklärt, dass Mitarbeiter die Daten zur Analyse und Verbesserung des Systems bekommen haben. Die Daten wurden allerdings auch in Gruppenchats und privat verbreitet und sich über private Situationen lustig gemacht. Die Nutzer hatten zwar zugestimmt, dass diese Daten aufgezeichnet und zur Verbesserung der Systeme genutzt werden, aber nicht dazu, dass diese Daten genutzt werden, um sich über sie lustig zu machen oder sie an andere weitergeben werden. Durch diesen Artikel wird klar, dass die Privatsphäre der Nutzer auch vom Hersteller eines CVs beeinträchtigt werden kann. Daher ist es wichtig, auch Möglichkeiten für den Schutz der Privatsphäre für diesen Fall zu berücksichtigen.

1.1 Behandeltes Problem

In Anbetracht des zuvor genannten Problematik wird eindeutig, dass es nötig ist Techniken zu finden, die den Schutz der Privatsphäre verbessern. Dazu wurden bereits einige Techniken entworfen, die als Privacy Enhancing Technologies (PETs) bezeichnet werden. PETs sollen die Privatsphäre von Nutzern zu schützen, ohne dabei die Nutzung der gewollten Dienste zu verhindern. Wie genau sie definiert sind, wird in Abschnitt 2.1 beschrieben. Jedoch gibt es einige Punkte, die beachtet werden müssen.

Oft muss ein Kompromiss zwischen dem Grad des Schutzes und dem Umfang des Komforts, beziehungsweise der Funktionalität des Fahrzeuges, gefunden werden. Dabei kann es sein, dass unterschiedliche Personen bereit sind unterschiedlich stark auf ihren Komfort zu verzichten um ihre Privatsphäre zu schützen. In der Studie von Zhou et al. [ZLMZ22] sieht man, dass Befragte unterschiedlich stark dazu bereit sind, ihre persönlichen Daten, für die Entwicklung neuer Produkte, zu teilen. Außerdem kann nicht nur eine Technik entwickelt werden, die für jeden Anwendungsfall und jeden Datentyp verwendet werden kann. Das liegt daran, dass es viele Datentypen und Anwendungsfälle gibt, die je nach den Anforderungen des Nutzers ausgewählt werden können. Diese Anwendungsfälle werden in Abschnitt 4.5 beschrieben. Des Weiteren hat der Aufbau der Infrastruktur hat einen Einfluss auf den Aufbau der Techniken. Je nachdem vor wem genau die Daten geschützt werden sollen, müssen andere Techniken verwendet werden.

Des Weiteren gibt es PETs, die nicht speziell für den Einsatz in Fahrzeugen erstellt wurden, dort jedoch auch verwendet werden können. So beispielsweise das von Sun et al. [SCZ20] entwickelte MicShield. Dieses soll persönliche Gespräche vor dem Mithören durch einen Personal Voice-Assistant (PVA) schützen. Dazu wird das Mikrofon so gestört, dass das Gesagte nicht verstanden werden kann. Wird allerdings das Wort zur Aktivierung des PVA gesagt, so wird die Störung unterbrochen und es kann ein Befehl gegeben werden. Diese Technik kann auch in Fahrzeugen eingesetzt werden, da hierfür lediglich das Wort zur Aktivierung angepasst werden muss. Auch diese nicht speziell für Fahrzeuge entwickelten Techniken sind schwer zu finden, da es Expertise benötigt herauszufinden, ob und wie sie im Fahrzeug eingesetzt werden können. Des Weiteren können einige der Techniken zwar nicht direkt im Fahrzeug umgesetzt werden, sondern benötigen Anpassungen, wodurch sie für den Endnutzer wiederum schwer zu finden sind.

Diese Arbeit soll eine Lösung für die Verbesserung der Privatsphäre bieten. Da es wie beschrieben nicht eine Lösung für alle gibt, gibt es eine große Anzahl an möglichen PETs, die auch aus anderen Bereichen übernommen werden können. Durch die zuvor beschriebenen Problematiken, wie die nötigen Anpassungen der PETs, die sich für den Endnutzer ergeben, ist es für ihn schwierig ein passendes PET für seine Zwecke zu finden. Dadurch ist es ihnen nicht möglich, sie einzusetzen und damit ihre Privatsphäre zu schützen. Dieses Problem soll in dieser Arbeit behandelt werden.

1.2 Lösungsvorschlag

Um das zuvor genannte Problem zu lösen wurde im Zuge dieser Arbeit ein System, mit dem Namen **Pri**vacy **T**echnologie **Ex**change (PriTeX), entworfen, in welchem die PETs gespeichert und gefunden werden können. Dazu wurde zunächst ein Konzept entworfen, welches das speichern und suchen der PETs ermöglicht. Danach wird eine Implementierung dieses Konzepts gegeben. Dem System

wurde der Name PriTeX gegeben. Es besteht aus drei Komponenten. Einem Datenspeicher oder „Repository“, einer berechnenden Komponente oder „Backend“ und einer graphischen Oberfläche. Diese Komponenten verwenden ein Metadatenmodell für die Speicherung der Eigenschaften der PETs und der Suche nach diesen PETs. Zusammen können die Komponenten und die Speicherung und Suche nach den PETs umsetzen.

Da das Metadatenmodell von den anderen Komponenten verwendet wird und dadurch der Kern von PriTeX darstellt, wird das Augenmerk auf das Metadatenmodell und damit die Suche nach möglichen PETs, gelegt. Für die anderen Komponenten wird ein Vorschlag für die Implementierung gegeben. Für das Metadatenmodell wird ein Schema entworfen, mit welchem die Metadaten gespeichert werden können. Mit der Hilfe der gespeicherten Metadaten kann dann nach den PETs gesucht werden.

In Kapitel 2 werden die Voraussetzungen und Anforderungen, die an PriTeX gestellt werden, beschrieben. Als Nächstes wird in Kapitel 3 das entwickelte Konzept von PriTeX erklärt, gefolgt von einer Beschreibung des Konzepts des entwickelten Metadatenmodells. Danach folgt in Kapitel 5 eine Beschreibung der Implementierung für das Metadatenmodell. In Kapitel 6 wird die Umsetzung der zuvor definierten Anforderungen evaluiert. Dann folgt eine Darstellung von verwandten Arbeiten in Kapitel 7. Zuletzt wird in Kapitel 8 noch eine Zusammenfassung und ein Ausblick gegeben.

2 Voraussetzungen und Anforderungen

Zuvor wurde bereits beschrieben, welches Problem in dieser Arbeit gelöst werden soll. Um eine gute Lösung zu finden, müssen zunächst die Voraussetzungen definiert werden. Dazu werden in Abschnitt 2.1 zunächst die PETs definiert, um so die zu berücksichtigenden PETs zu beschränken. Anschließend wird die angenommene Architektur der CVs in Abschnitt 2.2 dargestellt. Für das Lösen des Problems soll PriTeX entworfen werden, durch welches PETs gespeichert und gefunden werden können. Für dieses ist es nötig, die Anforderungen zu beschreiben, was in Abschnitt 2.3 getan wird.

2.1 Definitionen für Privacy Enhancing Technologies (PET)

Zunächst ist es wichtig, zu definieren, was ein PET ist. In der Literatur gibt es noch keine einheitliche Definition, daher wird hier eine eigene entworfen. Diese soll zu den bisher verwendeten Definitionen passen, weshalb drei von diesen zunächst genannt werden.

Definition 2.1.1 (Privacy enhancing technology [Fis09])
„Privacy-Enhancing Technologies (PETs) can be defined as technologies that are enforcing privacy principles in order to protect and enhance the privacy of users of information technology (IT) and/or of individuals about whom personal data are processed (the so-called data subjects).“

Übersetzung: Privatsphäreverbessernde Technologien (PETs) können als Technologien definiert werden, welche die Privatsphäreprinzipien durchsetzt, um die Privatsphäre, von Nutzern von Informationstechnologien (IT), zu schützen und verbessern und/oder von Individuen, von welchen persönliche Daten verarbeitet werden (sogenannte Datensubjekte).

Die erste Definition 2.1.1 wurde von Fischer-Hübner [Fis09] genutzt. In ihr werden Technologien, die die Privatsphäreprinzipien der Nutzer von Informationstechnologien umsetzen, als PET definiert. Der Nutzerkreis wird also auf Nutzer von sogenannten Informationtechnologien beschränkt. Dabei sind Informationstechnologien Technologien, die Daten verarbeiten. Diese definiert die Techniken sehr detailliert und verwendet bereits Fachbegriffe, wie Datensubjekt, wodurch sie für Endnutzer kompliziert zu verstehen ist. Da das System eine Definition aufführen soll, die für Endnutzer verständlich ist, damit dieser verstehen kann, welche Techniken durch PriTeX gespeichert und gefunden werden können, sollte eine leicht verständliche Definition verwendet werden, die keine Vorkenntnisse benötigt. Des Weiteren wird nicht auf den, in Abschnitt 1.1 genannten, Kompromiss zwischen Schutz der Privatsphäre und Funktionalität eines Services, der die gefilterten Daten verwendet. Dieser Kompromiss soll in der hier verwendeten Definition widergespiegelt werden.

Definition 2.1.2 (Privacy enhancing technology [UN23])
„Privacy-enhancing technologies (PETs) are technologies designed to safely process and share sensitive data.“

Übersetzung: „Privatsphäreverbessernde Technologien (PETs) sind Technologien, die dafür entworfen sind, sensible Daten sicher zu verarbeiten und zu teilen."

Diese zweite Definition 2.1.2 von den Vereinten Nationen [UN23] beschreibt PETs als Technologien, die sensible Daten sicher verarbeiten und teilen können. Durch die einfache Sprache ist sie auch für Endnutzer verständlich. Sie umfasst auch mehr als nur den Nutzer, da sie für sensible Daten definiert ist und die Person zu der diese Daten gehören nicht genauer definiert. Da manche der bereits entwickelten und als PETs bezeichnete Techniken aber auch andere Personen als nur den Nutzer schützen können, ist diese Definition zu knapp. Außerdem geht sie wie die erste nicht auf den Kompromiss zwischen Schutz und Funktionalität ein.

Definition 2.1.3 (Privacy enhancing technology Borking et al. [BVE+03])
„Privacy-Enhancing Technologies is a system of ICT measures protecting informational privacy by eliminating or minimising personal data thereby preventing unnecessary or unwanted processing of personal data, without the loss of the functionality of the information system."

Übersetzung: Privatsphäreverbessernde Technologien sind ein System von Informations- und Kommunikationstechnologie (ICT) Maßnahmen, welche die Informationsprivatsphäre schützen, indem die persönlichen Daten eliminieren oder minimieren und damit die unnötige oder unerwünschte Verarbeitung der personenbezogenen Daten vermeiden, ohne den Verlust der Funktionalität des Informationssystems.

Die dritte Definition 2.1.3 beschreibt PETs ein System von Maßnahmen, von Informations- und Kommunikationstechnologie, welche die informative Privatheit schützen, indem die sensiblen Daten minimiert oder eliminiert werden. Dabei sind Informations- und Kommunikationstechnologie die Technologien im Bereich von Information und Kommunikation. Durch die Minimierung oder Eliminierung der Daten soll vermieden werden, dass die Daten in einer unerwünschten Art verarbeitet werden. Die Definition spricht zudem nicht von einem Nutzer und nennt keine bestimmte Person, sondern es verallgemeinert, indem von personenbezogenen Daten gesprochen wird. Sie nennt außerdem, im Gegensatz zu den anderen Definitionen, auch den Kompromiss zwischen Schutz und Funktionalität. Diese Arbeit soll auf technologische Lösungen eingehen, die allein stehen und nicht ein Zusammenschluss mehrerer Maßnahmen sind. Auch ist diese Definition für einen Endnutzer schwer zu verstehen, da Fachbegriffe verwendet werden. Daher ist auch diese Definition nicht für die Anforderungen in dieser Arbeit verwendbar.

Aus den zuvor genannten Definitionen wird nun eine eigene Definition erstellt werden. Dabei sollen die Mangel, die in den vorherigen Definitionen festgestellt wurden, ausgebessert werden. Dadurch entsteht die folgende Definition.

Definition 2.1.4 (Privacy enhancing technology (Finale Definition))
„Privacy-enhancing technologies (PETs) sind Technologien, die es den Nutzern erlauben, ihre Privatsphäre und personenbezogenen Daten, sowie die anderer, zu schützen, wobei die Funktionalität von verwendeten Diensten erhalten bleiben soll."

Diese neue Definition 2.1.4 ist leicht verständlich und enthält wenige Fachbegriffe. Dadurch ist sie auch für Endnutzer verständlich und kann verwendet werden, um diesen die Grundlagen der PETs zu vermitteln. Außerdem wird auf alle wichtigen Aspekte, die unter PET verstanden werden soll eingegangen. Hier wird nicht nur auf die persönlichen Daten der Nutzer eingegangen, sondern ein

PET nach dieser Definition kann auch die persönlichen Daten von anderen Personen als dem Nutzer schützen. Auch wird auf den Kompromiss zwischen Schutz und Funktionalität eingegangen. Dadurch wird eine komplette Abschottung eines eigentlich vernetzten Fahrzeuges nicht als PET angesehen, auch wenn durch diese Methode die Privatsphäre des Fahrers und aller weiteren aufgezeichneten Personen geschützt wird. Denn durch die komplette Abschottung wird auch die Nutzung der durch die Vernetzung bereitgestellten Funktionen unmöglich, wodurch die Methode nicht als PET gewertet wird. Zu beachten ist hier, dass nicht weiter definiert ist, wie diese Technologien aussehen. Daher können es sowohl Software als auch Hardware Lösungen sein. In dieser Arbeit liegt das Augenmerk auf den Softwarelösungen, da diese im Gegensatz zu den Hardwarelösungen digital gespeichert und weitergegeben werden können.

2.2 Angenommene Architektur des Vernetzten Fahrzeugs (CV)

Wie bereits etabliert, sollen im entwickelten System PriTeX Technologien gespeichert werden, die die Privatsphäre schützen können. Aus diesem Grund ist es wichtig zu verstehen, vor wem man sich der Nutzer schützen will. Dazu muss identifiziert werden, wie das CV mit der Cloud kommuniziert und wer Zugriff auf die geteilten Daten hat. Im Bereich der CVs gibt es verschiedene mögliche Architekturen und Bestandteile dieser Architekturen. Um die daraus entstehende Komplexität zu reduzieren, wird im Folgenden eine Architektur spezifiziert, die in dieser Arbeit angenommen wird.

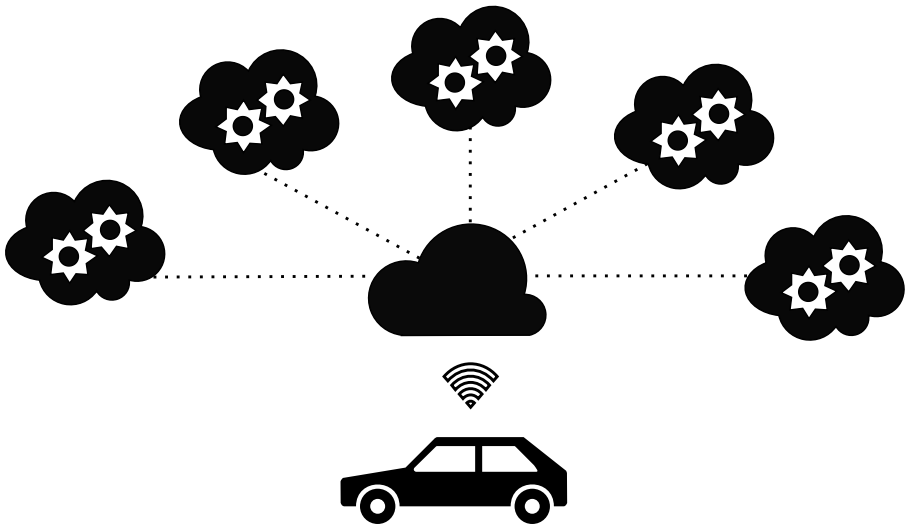


Abbildung 2.1: Typische Architektur für Connected Vehicles. Unten das Fahrzeug, in der Mitte der Hersteller und außen Drittanbieter.

Durch die Sensoren im CV werden die verschiedenen Datentypen und Daten aufgezeichnet. Das CV kann einfache Berechnungen selbst ausführen. Für kompliziertere Berechnungen und erweitere Funktionen werden die Daten an den Cloudservice des Herstellers weitergegeben. Stellt dieser die Funktion nicht selbst zur Verfügung, gibt er die Daten an Drittanbieter weiter. Das Ergebnis wird dann wieder an das CV zurückgegeben. Inwieweit die Daten zensiert oder gefiltert werden, hängt dabei von der Anwendung ab. Diese Architektur kann in Abbildung 2.1 angesehen werden.

Ist an der Berechnung nur das Fahrzeug selbst beteiligt und es werden keine Daten weitergegeben, wird das hier als eine lokale Berechnung bezeichnet. Da dabei keine Daten ausgetauscht werden, besteht auch keine Gefahr für die Privatsphäre. Dadurch ist das eine simple Möglichkeit für den Nutzer, seine Daten zu schützen. Dann ist der Einsatz von PETs nicht nötig. Wie bereits etabliert, ist dies aber nicht immer oder nur mit Einschränkungen der Funktionen möglich. Für den Einsatz von PETs wird hier angenommen, dass das CV wie der Name schon sagt, immer mindestens mit seinem Hersteller kommuniziert. Diese Kommunikationsart wird ohne Drittanbieter durchgeführt und deswegen auch als Berechnung ohne Drittanbieter bezeichnet. Sobald eine weiterer Anbieter Zugriff auf die Daten bekommt, wird das als Berechnung mit Drittanbieter bezeichnet.

Für die Berechnung von PETs ist dabei von Bedeutung, welchen Bestandteilen der Architektur man vertraut. So kann ein Nutzer beispielsweise dem Hersteller vertrauen, aber den Drittanbietern nicht. Ein anderer Nutzer traut aber auch dem Hersteller seines CV nicht. Vertraut ein Nutzer einem Bestandteil der Architektur, so bedeutet das hier, dass er bereit ist, seine Daten ungefiltert an diese weiterzugeben. Hier wird dabei davon ausgegangen, dass ein PET immer auf genau einem der Bestandteile der Architektur ausgeführt wird.

2.3 Anforderungen aus dem Data Mesh

Das Ziel für PriTeX ist es, wie bereits erwähnt, einem Nutzer die Suche nach PETs zu vereinfachen und die PETs zu speichern. Diese Anforderungen an das System sind allerdings sehr grob und müssen im folgenden genauer definiert werden. Dazu werden die Anforderungen für ein Data Mesh hinzugezogen, welche von Dehghani [Deh22] beschrieben werden. An diese werden Anforderungen definiert, die die Usability der Datendomäne sicherstellen sollen.

Zwar wird in dieser Arbeit kein Data Mesh entwickelt, dennoch sollen die im System enthaltenen PETs, wie auch die im Data Mesh enthaltenen Daten, die sogenannten Datenprodukte (engl. data product), nutzbar und nützlich sein. Ein Data Mesh soll Daten verteilt speichern, verwalten und zugänglich machen. Auch PriTeX soll etwas speichern, verwalten und zugänglich machen. Allerdings sollen es nicht Daten und Datenprodukte sein, sondern es sollen PETs gespeichert, verwaltet und zugänglich gemacht werden. Daher übernimmt PriTeX die Position des Data Mesh, wobei die PETs die Datenprodukte darstellen. Diese Zuordnung ist in Abbildung 2.2 dargestellt. Ein weiterer Unterschied, abgesehen von den gespeicherten Elementen, stellt der Ort der Speicherung dar. Im Data Mesh sind die Daten verteilt, während die PET zentral gespeichert, verwaltet und zugänglich gemacht werden. Durch diese Gemeinsamkeiten und Unterschiede können einige der Anforderungen übertragen werden, während andere nicht anwendbar sind. Im Folgenden werden diese Anforderungen erklärt und wenn möglich die Anwendung in PriTeX gegeben. Dabei werden zu Beginn die direkt übertragbaren Anforderungen genannt, gefolgt von denen, die abgewandelt werden müssen. Zuletzt werden die Anforderungen genannt, bei denen eine Übertragung nicht möglich oder sinnvoll ist.

Anforderung 1: Auffindbarkeit. Zunächst sollen Datenprodukte *auffindbar (engl. discoverable)* sein. Das bedeutet, dass es möglich sein soll, die gegebenen Daten zu überblicken, aber auch nach Daten zu suchen und gewünschte zu finden. Dadurch soll der erste Schritt bei der Nutzung des Systems vereinfacht werden, in welchem es darum geht, sich einen Überblick über das System zu verschaffen. Außerdem soll es möglich sein herauszufinden, welche Aktionen möglich sind und was diese bewirken. In dieser Arbeit sollen also die PETs im System leicht zu finden und

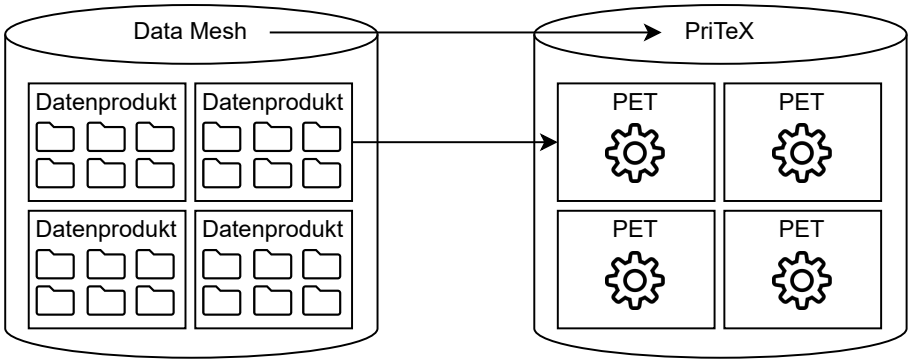


Abbildung 2.2: Mapping der Bestandteile des Data Mesh auf Bestandteile von PriTeX

zu überblicken sein. Auch soll ihre Bedeutung deutlich gemacht werden. Zudem soll das System intuitiv aufgebaut sein und einen bei diesem Prozess unterstützen. Dazu soll PriTeX eine Übersicht über die enthaltenen PET geben. Das kann beispielsweise mit einer graphischen Oberfläche erreicht werden.

Anforderung 2: Adressierbarkeit. Die zweite beschriebene Anforderung befasst sich mit der *Adressierbarkeit (engl. addressability)* des Datenprodukts. Hier wird beschrieben, dass es eine eindeutige Adresse geben soll, welche Rücksicht auf mögliche Änderungen des Datenproduktes nimmt. Solche Änderungen können das Hinzufügen von Daten oder auch semantische oder syntaktische Änderungen des Datenprodukts sein. Dazu soll das hier entwickelte System eine Adressierung der PETs erhalten. Diese Adressierung soll auch dann noch eindeutig sein, wenn ein PET sich verändert, also beispielsweise eine neue Version des PETs herausgebracht wird oder Änderungen an Syntax und Semantik gemacht werden. Dazu soll eine Beschreibung der PETs entwickelt werden, die diese Eigenschaften mit sich bringt.

Anforderung 3: Verständlichkeit. Die dritte Anforderung ist, dass das Datenprodukt *verständlich (engl. understandable)* sein soll. Damit ist gemeint, dass der Aufbau des Datenproduktes und die Bedeutung der darunter liegenden Daten verständlich ist. Es soll also nachvollziehbar sein, welche Daten das Datenprodukt enthält und wie diese zusammenhängen. Aber auch die Vorgänge innerhalb eines Datenproduktes sollen verständlich sein. Für PriTeX heißt das, es soll möglich sein, die Funktionsweise der PETs zu verstehen. Dazu gehört es, die benötigte Eingabe und die zu erwartende Ausgabe zu verstehen. Auch soll klar sein, wie stark die Einschränkung auf die vom CV bereitgestellten Funktionen durch den Einsatz des PETs, sein werden. Hier kann wie bei der ersten Anforderung eine graphische Oberfläche verwendet werden, um die Vorgehensweise der PET zu erklären.

Nachdem zuvor die Anforderungen beschrieben wurden, welche direkt anwendbar sind, wird im folgenden auf die Anforderungen eingegangen, die nicht direkt übernommen werden können. Dabei wird auch beschrieben, wie die Anforderungen mit der Hilfe leichter Anpassungen auch für PriTeX angewendet werden können.

Anforderung 4: Vertrauenswürdigkeit und Ehrlichkeit. Als vierte Anforderung soll das Datenprodukt *vertrauenswürdig und ehrlich (engl. trustworthy and truthful)* sein. Der Nutzer soll dabei das Vertrauten haben, dass die im Datenprodukt vorkommenden Daten auch der Realität entsprechen. Dabei geht es darum, wie viel vertrauen auf die aus den Daten abgeleiteten Folgerungen gelegt

werden kann. Auch soll der Nutzer erkennen können, ob das Datenprodukt für seine Anforderungen ausreichend ist. Für PriTeX kann diese Anforderung nicht direkt übernommen werden, da wir kein Äquivalent für die Daten im Datenprodukt haben. Stattdessen soll Vertrauen darin aufgebaut werden, dass ein gespeicherten PET von einer vertrauenswürdigen Quelle bereitgestellt werden und nicht verändert wurden. Dadurch soll etabliert werden, dass die PETs auch tatsächlich den Schutz bieten, den sie versprechen. Um das zu garantieren, kann PriTeX für die PETs einen Hash speichern, durch welchen überprüft werden kann, dass ein PET unverändert ist.

Anforderung 5: Sicherheit. Für die fünfte Anforderung soll das Datenprodukt *sicher (engl. secure)* sein. Das bedeutet, dass es möglich sein muss, den Zugang zum Datenprodukt sicher zu gestalten und zu kontrollieren wer auf die Daten zugreift. Dabei können sich die zugehörigen Richtlinien über die Zeit verändern. Auch ist der Zugriff auf die Daten auch facettenreich. So darf man beispielsweise die Daten nur ansehen, sie aber nicht veränder oder löschen. Das entworfene System soll also steuern, wer Zugriff auf die PETs hat. Dabei soll es darauf Rücksicht nehmen, wer PETs hinzufügen, verändern oder löschen darf. Um das zu verwirklichen, können Zertifikate an vertrauenswürdige Quellen vergeben welchen, und nur diesen das Einfügen neuer PETs oder das Ändern vorhandener PETs erlaubt werden.

Anforderung 6: Native Zugänglichkeit. Um das Datenprodukt in verschiedenen Anwendungen und von verschiedenen Personen nutzbar zu machen, wird die sechste Anforderung aufgeführt. In dieser wird definiert, dass das Datenprodukt *nativ zugänglich (engl. natively accessible)* sein soll. Dabei soll es einer großen Anzahl von unterschiedlichen Nutzern möglich sein, das Datenprodukt zu benutzen. Dazu soll das Datenprodukt mit den für den Nutzer nativen Tools nutzbar sein. Für das entwickelte System sollen die PETs in den CVs also nativ zugänglich sein. Das bedeutet, dass es für einen Nutzer möglich sein soll, ein gefundenes PET auch verwenden zu können. Dafür sollten die gespeicherten PETs in modularer Form vorliegen und ihr Interface generisch sein, sodass sie auf verschiedenen Fahrzeugen ausgeführt werden können.

Anforderung 7: Für sich allen stehend wertvoll. Die siebte Anforderung spricht davon, dass das Datenprodukt *für sich allein stehend wertvoll (engl. valuable)* sein soll. Dadurch soll sichergestellt werden, dass vor der Implementierung eines Datenprodukts klar ist, ob man diese auch benötigt und keine nicht gebrauchten Datenprodukte entwickelt werden. Dabei soll darauf geachtet werden, dass durch ein Datenprodukt immer ein Mehrwert entsteht und es ohne die Verbindung zu anderen Datenprodukten Wert hat. In dem entworfenen System sollten also nur PETs enthalten sein, die dem Nutzer einen Mehrwert bieten. Auch sollten sie diesen Mehrwert immer allein und ohne die Hilfe anderer PETs haben. Durch den Einsatz eines PETs soll also tatsächlich der Schutz der Privatsphäre verbessert werden. Wie zuvor in Abschnitt 2.1 definiert, sind PETs Technologien, die einen Schutz für die Privatsphäre bieten. Das bedeutet, dass sie schon der Definition nach, in unserem Sinne, von sich aus wertvoll sind, da eine Technologie, die keinen Schutz bietet, auch kein PET ist. Daher ist die Anforderung gegeben und kann im Folgenden vernachlässigt werden.

Die bisher beschriebenen Anforderungen für ein Data Mesh sind diejenigen, die auch für PriTeX angewendet werden können. Zu diesen wurden die nötigen Anpassungen erklärt. Doch nicht alle Anforderungen, die an ein Data Mesh gestellt werden, können auch für PriTeX verwendet werden. Im Folgenden wird diese nicht übertragbare Anforderung beschrieben.

Anforderung 8: Interoperabilität. Die achte und letzte Anforderung ist es, das Datenprodukt *interoperabel (engl. interoperable)* zu machen. Hier geht es darum, dass es möglich sein soll, Daten über mehrere Gebiete hinweg verarbeiten und kombinieren zu können. Dazu ist es wichtig, dass einige

Dinge der Datenprodukte, wie beispielsweise Schemas oder Metadatenfelder, standardisiert sind. Wie in der siebten Anforderung ist auch hier keine direkte Übertragung möglich. Interoperabilität ist eine Eigenschaft, die für heterogene oder verteilte Systeme gelten kann. Da wie etabliert PriTeX allerdings ein zentrales System ist, ist es nicht sinnvoll, die Anforderung der Interoperabilität auf dieses anzuwenden. Sie wird im folgenden daher nicht weiter behandelt werden.

3 Konzept von PriTeX

Nachdem im vorherigen Kapitel die Voraussetzungen und Anforderungen beschrieben wurden, befasst sich dieses Kapitel mit dem daraus entstehenden Aufbau von PriTeX. Dabei wird das Konzept aus den Anforderungen hergeleitet. Dafür werden zunächst die nötigen Komponenten des Systems, mithilfe der zuvor definierten Anforderungen, identifiziert. Dabei wird außerdem beschrieben, wie die gefundene Komponente die in Abschnitt 2.3 beschriebenen Anforderungen erfüllen kann. Danach wird in eine Übersicht gegeben, in der die einzelnen Komponenten des Systems erklärt werden.

Es wurde bereits etabliert, dass mit der Hilfe von PriTeX ermöglicht werden soll, PETs zu speichern und zur Verfügung zu stellen. Für das Speichern von PETs würde ein einfacher Datenspeicher, auch Repository genannt, genügen. Da das hier entwickelte System digital sein soll, liegt das Augenmerk auf den Softwarelösungen. Deren Implementierungen könnten dort abgelegt und der Zugriff auf sie ermöglicht werden. Die Hardwarelösungen können auf der anderen Seite nicht in einem digitalen System gespeichert werden oder durch dieses weitergegeben werden. Daher werden sie im folgenden nicht betrachtet. Das Repository kann zudem den nativen Zugriff (A6) umsetzen. Dazu können unterschiedliche Implementierungen und Versionen eines PET gespeichert werden, wodurch sichergestellt werden kann, dass sie für unterschiedliche Fahrzeuge verwendet werden können. Dabei sollte immer mindestens eine Implementierung für ein PET vorhanden sein. Das Repository stellt also den insgesamt den Datenspeicher für das System dar. Da aber noch andere Anforderungen definiert wurden, muss PriTeX aus mehreren Komponenten aufgebaut sein.

Um die Anforderungen der Auffindbarkeit (A1) umsetzen zu können, muss mit dem Nutzer interagiert werden. Für diese Interaktion eignet sich eine graphische Oberfläche, mit welcher die Anschaulichkeit gesteigert werden kann. Dort kann eine Übersicht über die gespeicherten PETs gegeben und die ihre Eigenschaften erklärt werden. So kann sichergestellt werden, dass der Nutzer die PETs finden kann. Außerdem ist es möglich eine verständliche Beschreibung zu geben, durch welche die Verständlichkeit (A3) der PETs erhöht werden kann. Für die Suche sendet die graphische Oberfläche die Suchanfrage des Nutzers an und stellt dann die Antwort für den Nutzer dar.

Die zwei bisher gefundenen Komponenten, das Repository und die graphische Oberfläche, können einige der Anforderungen umsetzen. Mit diesen Komponenten können die PETs gespeichert und erkundet werden. Doch wir brauchen eine weitere Komponente, welche die Suche ermöglicht. Bevor diese jedoch beschrieben werden kann, ist es zunächst nötig eine Beschreibung der PETs zu finden, die die Suche ermöglicht. Diese Beschreibung sollte ein einheitliches Format haben. Dafür kann ein Metadatenmodell verwendet werde, mit welchem die gespeicherten PETs mit ihren wichtigsten Eigenschaften beschrieben werden kann. Durch das Metadatenmodell ist es möglich, mit einer weiteren Komponente, eine Suche nach bestimmten PETs durchzuführen. Dadurch wird die Auffindbarkeit (A1) erhöht. Des Weiteren kann die Adressierbarkeit (A2), dadurch umgesetzt werden. Dafür ist es nötig, dass das Metadatenmodell auch auf Änderungen der PET darstellen soll, um auch bei möglichen neuen Versionen noch eindeutig zu bleiben. Dann können die Metadaten eines

PET als seine Adresse angesehen werden. Auch die Vertrauenswürdigkeit (A4) der PETs, kann hier erhöht werden. Das ist durch ein Metadatenfeld für den zuvor erwähnten Hash möglich. Die genaue Umsetzung wird in Abschnitt 4.4 beschrieben. Für die Umsetzung der Sicherheit (A5) können, des Weiteren, Zertifikate verwendet werden. Wie genau diese Zertifikate in das Metadatenmodell eingefügt und verwendet werden, wird in Abschnitt 4.4 bechrieben. Dieses Metadatenmodell stellt keine eigene Komponente dar, sondern wird von den Komponenten von PriTeX genutzt.

Zuvor wurde bereits beschrieben das die zwei bisher identifizierten Komponenten nicht alle Anforderungen umsetzen können. Auch mit dem Metadatenmodell ist es nicht möglich, alle Anforderungen umzusetzen. Um die Suche durchzuführen, wird eine weitere Komponente verwendet. Diese wird im folgenden als Backend bezeichnet. Es nimmt außerdem die Suchanfragen des Frontends an und gibt die Antwort an dieses zurück. Dabei nutzt es das Metadatenmodell, um die im Repository gespeicherten Metadaten umzuwandeln und die Suche umzusetzen. Für die Suche können die im Metadatenmodell festgelegten Kategorien an PETs verwendet werden, um ein PET mit bestimmten Eigenschaften zu finden. Des Weiteren lässt sich Anforderung der Sicherheit (A5) nicht durch die beiden anderen Komponenten umsetzen. Dazu können Zertifikate eingesetzt werden, die einen Dienst oder Anbieter als vertrauenswürdig identifizieren. Diese Zertifikate können von einer anderen vertrauenswürdigen Quelle vergeben und auch überprüft werden. So bietet beispielsweise der TÜV eine Möglichkeit an, seine Dienste und Produkte zu zertifizieren [TÜV23]. Dazu braucht es eine Komponente, welche diese Berechnungen und die Zulassung zur Datenbank durchführen kann.

Insgesamt ergibt sich dadurch das in Abbildung 3.1 gezeigte Aufbau des Systems System. Es besteht aus dem Backend, dem Repository und der graphischen Oberfläche. Dort werden im Repository einige PETs dargestellt. Hierbei sollen die Zahnräder die Implementierungen repräsentieren. Das Listensymbol steht für die Metadaten des jeweiligen PET. Die Speicherung der Metadaten geschieht im Repository.

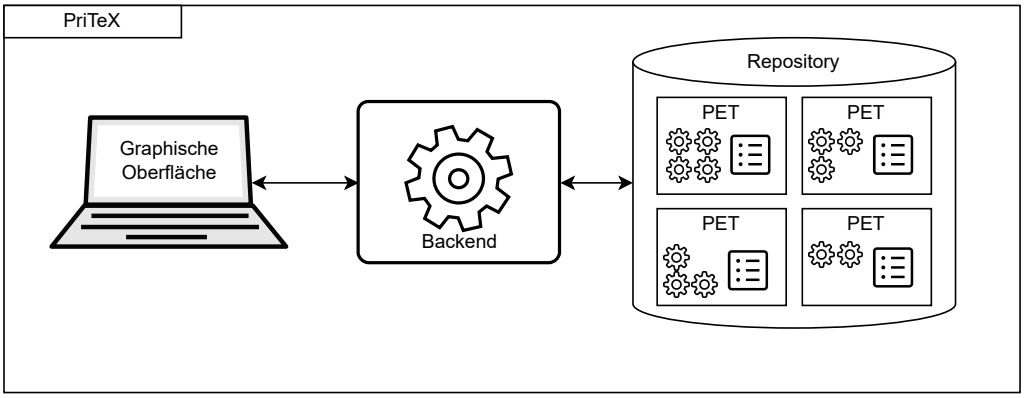


Abbildung 3.1: Konzeptioneller Aufbau des Systems mit einzelnen Komponenten.

Um die Anfrage eines Nutzers zu bearbeiten, arbeiten die Komponenten wie folgt zusammen. Der Nutzer kann seine Anfrage über die graphische Oberfläche eingeben. Dazu können in der graphischen Oberfläche Erklärungen für die Metadatenfelder gegeben werden und so die Einteilungen der PETs in Kategorien und die Funktionsweise demonstriert werden. Dadurch wird der Verständlichkeit (A3) der PETs erhöht. Die Anfrage wird dann an das Backend weitergeleitet. Das Backend verwendet dann die im Repository gespeicherten Metadaten der PETs, um ein passendes PET zu finden. Dabei orientieren sich die gespeicherten Metadaten und die Suche am Metadatenmodell. Nachdem

PETs gefunden wurden, die zur Anfrage des Nutzers passen, wird das Ergebnis der Suche an die graphische Oberfläche zurückgegeben. Diese kann dann die gefundenen PETs für den Nutzer darstellen.

Während der Bearbeitung verwenden alle drei Komponenten das Metadatenmodell, um eine Suche zu ermöglichen. Es ist also das Kernstück von PriTeX. Daher werden im folgenden, zur Reduzierung der Komplexität der Implementierung, die einzelnen Komponenten nicht implementiert. Zu diesen gibt es bereits zahlreiche Möglichkeiten und Implementierungen, weswegen sie das hier nicht erneut durchgeführt wird. Stattdessen wird das Augenmerk auf das Metadatenmodell gelegt. Zu diesem gibt es bisher keine Umsetzung. Die Beschreibung des Metadatenmodells wird in Kapitel 4 gegeben.

4 Konzept des Metadatenmodells

Dieses Kapitel beschäftigt sich mit dem Metadatenmodell, welches von den drei Komponenten von PriTeX verwendet wird. Das Metadatenmodell soll, wie beschrieben, die Suche nach PETs ermöglichen. Dabei soll zunächst definiert werden, was ein „passendes“ PET sein kann. Dazu sollen verschiedene Arten von Anfragen ermöglicht werden. Des Weiteren wurden bereits die Anforderungen beschrieben, die durch dieses umgesetzt werden sollen. Zum einen soll das Metadatenmodell die Auffindbarkeit (A1) eines PETs erhöhen. Dazu soll es beschreiben, wen ein PET schützt, was genau, also welche Daten, es schützt, wo es ausgeführt werden kann und wie genau es schützt. Dadurch kann auch die Verständlichkeit (A3) eines PETs verbessert werden, da die wichtigsten Eigenschaften der PETs beschrieben werden. Dazu sollte allerdings die graphische Oberfläche mit eingesetzt werden. Durch das Metadatenmodell soll es außerdem möglich sein, die in PriTeX gespeicherten PETs zu adressieren (A2). Die gewählten Metadaten sollten die PETs also möglichst eindeutig beschrieben. Des Weiteren sollen die Metadaten die Vertrauenswürdigkeit (A4) und Sicherheit (A5) einer Implementierung etablieren können. Daher soll es möglich sein, zu verifizieren, ob eine Implementierung verändert wurde und wer Zugriff auf die Implementierung hat.

Um das Metadatenmodell zu entwickeln wird zunächst, in Abschnitt 4.1 auf die Anfragen eingegangen, die durch PriTeX bearbeitet werden sollen. Dann werden, in Abschnitt 4.2, die allgemeinen Metadaten eines PETs beschrieben. Anschließend wird, in Abschnitt 4.3, auf bereits entwickelte PETs eingegangen. Aus diesen werden dann die nötigen Metadatenfelder abgeleitet. Dabei wird Rücksicht darauf genommen, ob diese Kategorien für das Szenario des CV geeignet sind und ob sie in PriTeX berücksichtigt werden sollen. Danach wird, in Abschnitt 4.4, auf die Metadaten der Implementierungen eingegangen. Danach wird in Abschnitt 4.5 das, aus den identifizierten Metadatenfeldern, entwickelte Metadatenmodell vorgestellt und erklärt.

4.1 Unterstützte Anfragen

Die Aufgabe von PriTeX ist es, dem Nutzer die Suche nach einem „passenden“ PET zu ermöglichen und somit die Auffindbarkeit (A1) der PETs umzusetzen. Dabei können die Anforderungen, die ein Nutzer an ein PET stellt, je nach Nutzer unterschiedlich sein. Diese Anforderungen und daraus entstehenden Anfragen können in unterschiedliche Typen unterteilt werden. Ein PET, dass die Anforderungen des Nutzers erfüllt und durch eine bestimmte Art von Anfrage gefunden werden, ist dabei ein „passendes“ PET für diesen Nutzer. Hier werden drei Typen von Anfragen behandelt.

Typ 1: Suche nach einem PET nach den Daten, die geschützt werden sollen. Bei diesem Typ der Anfrage wird davon ausgegangen, dass der Nutzer bereits weiß, welche Daten er schützen möchte. Da ein PET meist nur einen bestimmten Datentyp schützen kann und nicht direkt auf andere Typen von Daten übertragen werden kann, kann so ein PET gefunden werden, dass bestimmte Daten schützen kann.

Typ 2: Suche nach einem PET nach dem Ausführungsort. Diese Anfragen sind für Nutzer, die ein PET finden möchten, welches auf ihrem Fahrzeug ausgeführt werden kann. Dazu muss die Anfrage die PETs ausschließen, deren Eigenschaften sie für die Ausführung im Fahrzeug unbrauchbar machen. Die Anfrage ist auch Nutzer nutzbar, die absichtlich nach einem PET suchen, welches außerhalb ihres Fahrzeuges ausgeführt werden muss. Die möglichen Ausführungsorte werden genauer in Abschnitt 4.2.1 beschrieben.

Typ 3: Suche nach einem PETs nach der Funktion, die genutzt werden soll. Nutzer können diese Anfragen verwenden, um nach einem PET zu suchen, welches die sensiblen Daten schützt, aber auch die Nutzung eines bestimmten Dienstes oder einer Funktion weiterhin zulässt. Dabei wird davon ausgegangen, dass ein Nutzer eine bestimmte Funktion nutzen möchte und nun nach einem PET sucht, mit dem er gleichzeitig seine Daten schützen kann.

Diese drei Anfragen stellen die voraussichtlich am häufigsten verwendeten Anfragen der Nutzer dar. Daher werden sie im Folgenden beachtet. Um diese Anfragen bearbeiten zu können, müssen die im Metadatenmodell enthaltenen Felder bestimmte Eigenschaften der PETs durch diese beschrieben werden. Zunächst müssen, um die PETs finden zu können, generelle Informationen des PETs gespeichert werden. Dadurch soll ein PET eindeutig identifiziert werden können und seine wichtigen Eigenschaften beschrieben werden. Die Metadatenfelder, die dafür benötigt werden, werden in Abschnitt 4.2 erläutert. Des Weiteren ist es wichtig, die spezifischen Eigenschaften zu beschreiben, die ein PET hat, basierend auf den Daten eines bestimmten Datentyps, den es schützen kann. Durch diese Beschreibung kann ein PET für bestimmte Anforderungen gefunden werden. Diese Eigenschaften und wichtigen Aufteilungen können aus den bereits existierenden PETs abgeleitet werden, was in Abschnitt 4.3 getan wird. Insgesamt kann so ein PET eindeutig beschrieben werde, wodurch die Adressierbarkeit (A2) gegeben wird. Zuvor wurde bereits in Kapitel 3 beschrieben, dass zu einem PET auch mehrere Implementierungen gespeichert werden können. Um die Adressierbarkeit (A2) auch für die Implementierungen zu gewährleisten, muss auch für diese eine eindeutige Beschreibung für diese durch das Metadatenmodell möglich sein. Diese Metadatenfelder werden in Abschnitt 4.4 beschrieben. Durch diese drei Arten von Metadatenfeldern sollen insgesamt wie beschrieben die Adressierbarkeit (A2) der PETs umgesetzt werden.

4.2 Allgemeine Metadaten

Nachdem zuvor, in Abschnitt 4.1, die Anfragetypen definiert wurden, sollen nun die Metadatenfelder identifiziert werden, die für die Umsetzung der Anfragen benötigt werden. Dazu sollen bereits existierende PETs betrachtet werden und aus ihnen Kategorien abgeleitet werden. Durch die Betrachtung der bereits existierenden PETs ist es möglich, ihre wichtigsten Eigenschaften herauszufinden. So können die wichtigen Unterscheidungen und Gemeinsamkeiten der PETs identifiziert werden. Dadurch kann sichergestellt werden, dass die PET sinnvoll adressiert (A2) und gesucht werden können.

Zunächst wird auf Kategorien eingegangen, die alle PETs gemeinsam haben. Die hier enthaltenen Metadaten sind unter anderem die allgemeinen Metadaten, die ein PET eindeutig identifizieren. Diese Metadatenfelder werden im Folgenden beschrieben.

Name. Zunächst hat jedes PET einen Namen. Dieser sollte frei wählbar sein und kann beispielsweise durch den Anbieter des PETs festgelegt werden. Um tatsächlich frei wählbar zu sein, wird nicht verlangt, dass er eindeutig ist.

Identifikator. Um ein PET eindeutig identifizieren zu können, muss es einen Bezeichner geben, der eindeutig ist. Da, wie zuvor beschrieben, der Name nicht eindeutig sein muss, braucht man dafür ein eigenes Feld. Dieses Feld wird als Identifikator oder kurz ID bezeichnet.

Datentyp. Um ein PET zu finden, dass bestimmte Daten schützen kann, ist es wichtig, den Typ der bearbeiteten Daten zu speichern. So können dem Nutzer die PETs vorgeschlagen werden, die die Daten schützen können, die er als sensibel ansieht.

Zeitaufwand. Der Zeitaufwand eines PETs beschreibt, wie schnell die Berechnung durchgeführt wird. Das ist wichtig, wenn ein Nutzer eine Funktion nutzen möchte, die Daten in Echtzeit verarbeitet. Ein PET, welches einen hohen Zeitaufwand hat, kann für solche Funktionen nicht eingesetzt werde, da sonst eine Verzögerung in der Ausgabe der Funktion entsteht. Um also die Anwendbarkeit eines PET prüfen zu können, muss der Zeitaufwand in den Metadaten gespeichert werden, was durch dieses Metadatenfeld möglich ist.

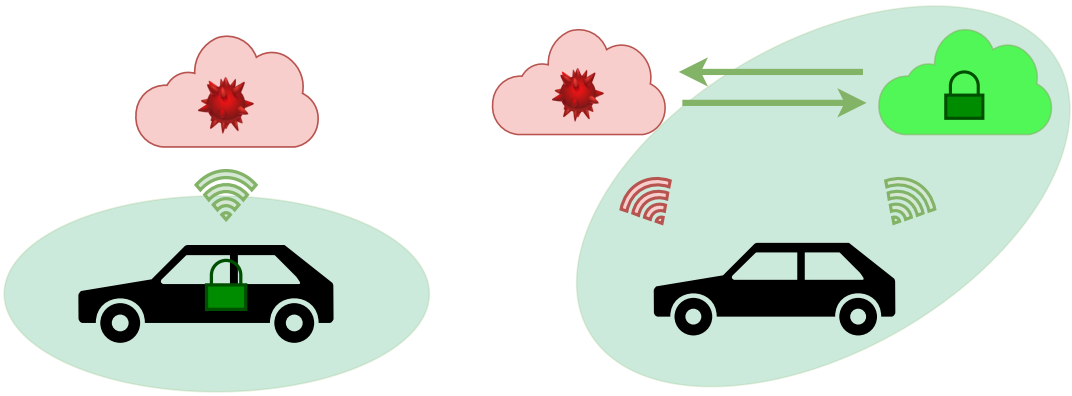
Um weitere Metadatenfelder der PETs zu identifizieren ist es wichtig, sich die Architektur, die ein PET verwendet anzuschauen. Das wird in Abschnitt 4.2.1 getan. Aus diesen möglichen Architekturen werden dann die nötigen Metadatenfelder abgeleitet.

Des Weiteren sind die betroffenen Personen von Bedeutung. Zwar kann nur der Fahrer, beziehungsweise Besitzer, tatsächlich Techniken im Fahrzeug einsetzen, dennoch können auch weitere Personen durch PETs geschützt werden. Die Personen, die durch ein CV betroffen sind und von diesem aufgezeichnet werden, werden in Abschnitt 4.2.2 beschrieben. Der zusätzliche Schutz für weitere Personen kann Einfluss auf die Entscheidung des Nutzers nehmen, sie einzusetzen. Daher müssen auch Metadaten gespeichert werden, die die geschützten Personen beschreiben können.

4.2.1 Nötige Architektur

Wie bereits in Abschnitt 2.2 beschrieben, stellt sich für den Ort der Berechnung von PETs die Frage, ob man einer weiteren Partei, außer dem eigenen Auto vertraut. Hier wird zwischen zwei möglichen Berechnungsorten unterschieden, welche in Abbildung 4.1 dargestellt sind. In den beiden Abbildungen (a) und (b) wird die Ausführung des PETs durch das grüne Schloss dargestellt. Die nicht vertrauenswürdige Partei wird durch die rote Wolke dargestellt. Der grüne Bereich gibt dabei den Bereich an, in welchem den beteiligten Parteien vertraut wird und in welchem die Daten ungefiltert verteilt werden können.

Zum einen können die Berechnungen der PETs auf den CV durchgeführt werden, was in Abbildung 4.1(a) dargestellt wird. So kann zum Beispiel die Aggregation, wie von Stach et al. [SGB+22] beschrieben, von Geschwindigkeitsdaten durch das Fahrzeug ausgeführt werden. Die Ausführung der PETs im CV hat den Vorteil, dass die Daten nicht an eine weitere Partei gegeben werden müssen. Daher muss man sich nicht entscheiden, welcher Partei man vertrauen kann oder möchte. Da auf einem CV jedoch nicht unbegrenzt Ressourcen zur Verfügung stehen, können nicht alle PETs dort ausgeführt werden. Benötigt ein PET beispielsweise mehr Speicherplatz, als ein CV besitzt, so kann dieses PET nicht verwendet werden. Eine weitere Art von PETs, die nicht direkt auf dem CV ausgeführt werden kann, sind Techniken, die Daten von mehr als einem CV verwenden. Für diese



(a) **(b)**

Abbildung 4.1: Mögliche Ausführungsorte für ein PET. In Abbildung (a) ist der Ausführungsort im CV und in Abbildung (b) in einer dritten Partei. Der grüne Bereich stellt dabei den Bereich dar, in dem die Daten sich ungefiltert befinden, also der Bereich, dem vertraut wird.

Techniken braucht es einen Ort, an dem die Daten gesammelt werden und dort berechnet werden, bevor die Ausgabe an die Datenquelle zurückgegeben wird. Verwendet ein PET nur die Daten eines CVs, so kann es auch auf dem CV ausgeführt werden.

In Abbildung 4.1(b) wird das PET hingegen in einer weiteren Partei ausgeführt, weswegen das Schloss in einer Wolke dargestellt wird. Diese berechnende Partei, die nicht das CV selbst ist, wird im folgenden als dritte Partei bezeichnet, da sie neben dem CV und der nicht vertrauenswürdigen Partei die dritte Partei darstellt. Dabei kann die dritte Partei beispielsweise der Hersteller sein, aber auch andere Parteien, denen der Nutzer vertraut und die unterschiedlich von der ist, vor der man sich schützen möchte. Möchte der Nutzer sich also vor dem Hersteller schützen, so kann er diesen nicht als dritte Partei verwenden. Es ist, durch den Einsatz einer dritten Partei möglich, größere Mengen an Ressourcen zur Verfügung zu stellen, da im Gegensatz zu einem CV beispielsweise keine Platzeinschränkung vorhanden ist. Auch die Techniken, die für die Ausführung auf dem CV nicht geeignet sind, können mithilfe der dritten Partei ausgeführt werden. So beispielsweise die Technik von Gedik und Liu [GL08], welche die Daten mehrerer Datenquellen für die Berechnung verwendet und daher wie zuvor beschrieben nicht auf den CV ausgeführt werden kann. Der Nachteil davon, eine dritte Partei für die Berechnung des PETs zu verwenden, ist also, dass eine Partei gefunden werden muss, der man vertrauen kann oder möchte.

Nachdem die Vor- und Nachteile der Ausführungsorte beschrieben wurden, werden im Folgenden die nötigen Metadatenfelder beschrieben, die für die Auswahl des Ausführungsortes benötigt werden.

Speicherbedarf. Wie zuvor beschrieben, nimmt der benötigte Speicherbedarf eines PETs Einfluss darauf, wo es ausgeführt werden kann. Daher wird die Information, wie hoch der Bedarf eines PETs ist, in den Metadaten gespeichert.

Rechenleistung. Ähnlich wie der Speicherbedarf, kann auch die Rechenleistung bestimmen, wo ein PET ausgeführt werden kann. Benötigt ein PET mehr Rechenleistung, als ein CV zur Verfügung stellen kann, so kann das PET nicht dort ausgeführt werden. Die benötigte Rechenleistung bestimmt also, wo ein PET ausgeführt werden kann, weswegen es wichtig ist, diese Information zu speichern.

Anzahl der Datenquellen. Der Einfluss der Anzahl der Datenquellen auf den Ausführungsort wurde zuvor bereits beschrieben. Dabei wird je ein PET als eine Datenquelle angesehen. Die Anzahl der Datenquellen, die ein PET erwartet, kann in diesem Metadatenfeld gespeichert werden.

4.2.2 Betroffene Personen

In der Umgebung eines CV entsteht für mehrere Personengruppen ein Eingriff in die Privatsphäre. Diese Personengruppen werden im Folgenden identifiziert. Danach folgt eine Beschreibung des Metadatenfeldes gegeben, welches beschreibt, welche Personengruppen durch ein PET geschützt werden können.

Insgesamt können drei grobe Personengruppen identifiziert werden. Die erste Personengruppe umfasst die Fahrer der CV. Für sie entsteht durch das eigene Fahrzeug und die Weitergabe der durch dieses aufgezeichneten Daten ein kontinuierlicher Eingriff in die Privatsphäre. Kann man ihm das Fahrzeug zuordnen, so kann man ihm auch alle vom Fahrzeug weitergegebenen Daten zuordnen. Diese Zuordnung kann beispielsweise durch den verwendeten Schlüssel oder das verbundene Smartphone geschehen. Dadurch kann eine große Menge an sensiblen Daten zu einzelnen Personen gesammelt werden. Die zweite Personengruppe beinhaltet die weiteren Insassen, beziehungsweise Mitfahren, im CV. Auch sie werden von nach innen gerichteten Sensoren aufgezeichnet und mögliche Daten weitergegeben. Für sie ist die Zuordnung der Daten auf eine einzelne Person nicht so trivial wie für den Fahrer, aber auch sie können durch bestimmte Merkmale zum Beispiel durch Tonaufnahmen identifiziert werden. Dazu kann beispielsweise die von Singh et al. [SSS19] entwickelte Technik verwendet werden, in welcher zunächst Merkmale der Stimme extrahiert werden und dann der Euklidsche Abstand, für die Identifikation der Stimme, verwendet wird. Die letzte Personengruppe sind die außenstehenden Personen, also Personen, die sich außerhalb des CV befinden. Diese Gruppe umfasst eine große Anzahl kleinerer Gruppen, wie Fahrer anderer Fahrzeuge und Fußgänger. Sie werden von den nach außen gerichteten Sensoren aufgezeichnet. Auch sie können beispielsweise durch Kameraaufnahmen identifiziert werden. Dazu kann beispielsweise die Technik von Liu et al. [LJJ+18] verwendet werden. Sie nutzen sowohl das Aussehen, als auch die Bewegungsinformationen, um eine Person in einem Video zu identifizieren.

Geschützte. Dieses Metadatenfeld beschreibt, welche Personen und ihre Daten, durch den Einsatz eines PET geschützt werden können. Dadurch kann herausgelesen werden, ob ein PET die für den Nutzer wichtigen Personengruppen schützen kann.

4.3 Datentyp basierte Metadaten

Zuvor wurden bereits die Metadatenfelder identifiziert, die jedes PET besitzt. Dadurch können die grundlegenden Eigenschaften der PETs beschrieben werden. Wie zuvor beschrieben können PETs jedoch nicht für einen beliebigen Datentyp verwendet werden, sondern nur für den Datentypen, für den sie entwickelt wurden. Das gilt beispielsweise für die Technik von Hoh et al. [HGXA07], welche

speziell für den Schutz von Ortsdaten entwickelt wurde und für beispielsweise für Stimmdaten nicht einsetzbar ist. Dazu gibt es zunächst ein Metadatenfeld, welches erwarteten Datentyp der Eingabe angibt.

Datentyp. Dieses Metadatenfeld gibt an, welche Art von Daten als Eingabe in das PET erwartet werden. Durch die Wahl des Wertes dieses Metadatenfelds wird bestimmt, welche anderen Metadatenfelder für ein PET vorhanden sind und welche Werte gewählt werden können. Diese weiteren Metadatenfelder sind also vom Datentypen abhängig.

Da ein Nutzer aber eine Anfrage von Typ 1 stellen könnte, bei der er bereits weiß, welche Daten er schützen möchte, ist es wichtig auch die PETs eines bestimmten Datentypen weiter einzuteilen. Für diese Arbeit wurden vier Datentypen ausgewählt, die exemplarisch zeigen sollen, wie sich die Kategorien innerhalb der Datentypen unterscheiden können. Die ausgewählten Datentypen stellen eine sehr große Gefahr für die Privatsphäre dar und werden deswegen als Erstes in PriTeX eingefügt. Zu diesen wichtigen Datentypen gehören die Ortsdaten, die Stimmdaten und die Kameradaten. Der vierte Datentyp, die Geschwindigkeitsdaten, stellen eine geringere Gefahr für die Privatsphäre und die Identifikation der involvierten Personen, dar. Da es sich aber um eindimensionale, numerische Daten handelt, werden sie hier als Platzhalter für andere Datentypen mit den gleichen Eigenschaften verwendet. Der Eingriff in die Privatsphäre, der durch das Teilen von Daten dieser vier Datentypen entstehen kann, wird zu Beginn jedes Abschnittes kurz eingeführt. Dabei entsteht die Reihenfolge der Kapitel aus der Komplexität der Datentypen. Daher wird zunächst auf die Geschwindigkeitsdaten, dann auf die Ortsdaten, die Stimmdaten und die Kameradaten eingegangen.

4.3.1 Geschwindigkeitsdaten

Für sich allein gesehen erscheinen Geschwindigkeitsdaten unwichtig für den Schutz der Privatsphäre. Es fällt schwer, sich einen tatsächlichen Eingriff in die Privatsphäre vorzustellen, der im Zusammenhang mit der Geschwindigkeit steht. Betrachtet man sie aber in Verbindung mit anderen Datentypen wird klar, dass auch durch diesen Datentyp Lücken in der Privatsphäre entstehen können. Vor allem in Anbetracht von Ortsdaten bedeuten sie ein Risiko. Dort kann man mithilfe ungefilterter Geschwindigkeitsdaten zum Beispiel Dummy-Objekte erkennen und filtern. Um das zu tun, kann man die Objekte ausschließen, die vom vorherigen Objekt aus mit der gegebenen Geschwindigkeit nicht zu erreichen sind oder zu nah sind. Dadurch können für Ortsdaten sichere Techniken, durch dieses Hintergrundwissen unsicher werden. Das wird durch das folgende Beispiel weiter klar.

Der Artikel vom Spiegel [Spi23] beschreibt ein solches Problem. Er berichtet davon, dass es möglich ist, ausgehend von einer Startposition und folgenden Geschwindigkeitsdaten, die Fahrstrecke eines Autos nachvollziehen zu können. Dabei werden die Geschwindigkeiten auf die gegebenen Straßenverhältnisse abgebildet. Wird man auf einer schnellen Straße langsamer und danach wieder schneller, kann man daraus unter den richtigen Bedingungen schließen, dass das Auto abgebogen ist. Es kann also die genaue Fahrstrecke berechnet werden, was einen großen Eingriff in die Privatsphäre darstellt.

Ein einfacher Weg, um die Geschwindigkeit zu verschleiern, besteht darin, eine bestimmte Geschwindigkeit festzulegen. Das ist jedoch sehr auffällig und kann schnell als Manipulation erkannt werden.

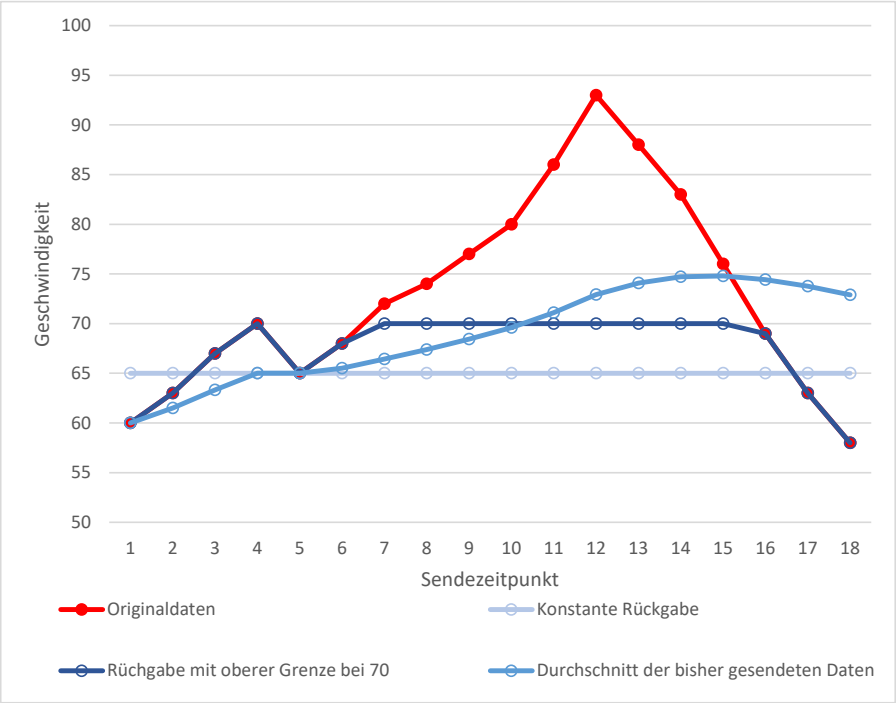


Abbildung 4.2: Beispiel für mögliche Verschleierungen

Eine weitere Idee ist es eine Höchstgeschwindigkeit festzulegen und dann alle Datenpunkte, die darüber liegen, auf diese Geschwindigkeit abzubilden. Doch auch das ist sehr auffällig. Bekommt man viele gleiche Geschwindigkeiten übertragen, kann man daraus lesen, dass diese beeinflusst wurden.

Auch viele andere Berechnungen, die auf eindimensionalen numerische Daten ausgeführt werden können, können für die Verschleierung verwendet werden. Einige dieser Techniken werden in Abbildung 4.2 beispielhaft dargestellt. Hier werden die originalen Daten durch die roten Datenpunkte und ihr Verlauf durch die rote Linie dargestellt. Die horizontale Achse zeigt, zu welchem Zeitpunkt ein Datenpunkt gesendet wurde, wobei keine genaue Einheit gegeben wird. Auch für die vertikale Achse wird keine genaue Einheit gegeben, da dort die Berechnungen von dieser unabhängig sind. Diese Achse beschreibt die Geschwindigkeit. Es kann zum Beispiel der Durchschnitt der bisher aufgezeichneten Geschwindigkeitsdaten weitergegeben werden, was durch die blaue Linie mittlerer Helligkeit gezeigt wird. Die zuvor beschriebene konstante Rückgabe wird durch die hellblaue beziehungsweise die Rückgabe in Grenzen durch die dunkelblaue Linie dargestellt. Insgesamt wird gezeigt, wie sich der Einsatz der Techniken auf die zurückgegebenen Daten auswirkt.

Durch Speed Anonymization (SpAn), beschrieben von Held [Hel22], wird versucht Geschwindigkeitsdaten zu anonymisieren, ohne ihre Glaubwürdigkeit zu beeinträchtigen. Dabei wird innerhalb einer höchsten und niedrigsten Geschwindigkeit eine zufällige neue Ausgabe generiert, welche glaubwürdig nahe an der vorherigen Ausgabe liegt. Insgesamt wird erreicht, dass die ausgegebenen Daten einen zusammenhängenden Geschwindigkeitsverlauf ergeben. Stellt man die Parameter richtig ein, ist es schwierig, die Ausgabe von einem echten Verlauf zu unterscheiden.

4.3.2 Ortsdaten

Eine häufig verwendete Funktion in CVs ist die Navigation an einen Zielort. Dabei wird, für das Berechnen der optimalen Route, auch das aktuelle Verkehrsaufkommen in Betracht gezogen. Dafür sendet das CV seinen aktuellen Standort an einen Server, der ihm die gebrauchten Informationen weiterleitet. Doch auch andere Funktionen, wie das Finden der nächstgelegenen Tankstelle, werden nur durch das Weiterleiten des Standortes möglich. Solche Services, die die Ortsdaten von Nutzern verwenden, werden auch als Location-based Services (LBSs) [SV04] bezeichnet.

LBSs umfassen einen großen Bereich an möglichen Funktionen. Abgesehen der bereits genannten Möglichkeiten, können LBS auch für viele weitere Funktionen genutzt werden. So können sie auch für Vorschläge für Restaurants oder Veranstaltungen genutzt werden oder auch in Videospielen zum Einsatz kommen. Dieser Einsatz in Videospielen wird von Xanthopoulos und Xinogalos [XX16] erläutert. Nicht für alle diese Funktionen sind die gleichen Aspekte der Ortsdaten notwendig. Aus diesem Grund gibt es unterschiedliche PETs, die diese unterschiedlichen Aspekte schützen, aber andere notwendige Aspekte der Ortsdaten beibehalten. Auch hier gibt es also nicht eine Lösung für alles. Daher bietet es sich an, die PETs in Gruppen einzuteilen, um besser zu verstehen, wofür sie zu verwenden sind. Jiang et al. [JLZ+21] teilen diese mithilfe von drei Metriken ein. Mit der Anwendungsart, der Architektur und den Methodenarten. Diese Aufteilung wird in Tabelle 4.1 dargestellt. Mit der Hilfe dieser Metriken ist es möglich, die PETs in Kategorien einzusortieren. Diese werden im folgenden genauer beschrieben. Dabei wird auch auf die Verwendung der Metriken in dieser Arbeit eingegangen. Da auf die Architektur, für die Berechnung von PETs, bereits in Kapitel 4.2.1 eingegangen wurde, wird das hier nicht erneut erläutert.

Anwendungsarten von Ortsdaten

Eine wichtige Aufteilung der Anwendung, von PET für Ortsdaten, stellt die Differenzierung nach der Häufigkeit der Aufnahmen dar. Hierbei wird zwischen zwei Klassen unterschieden. Die eine Klasse befasst sich mit Techniken für die Verschleierung von Momentaufnahmen des Standorts dar. Dieser Fall ist im Verhältnis zum zweiter weniger Komplex. Da hier immer nur ein Augenblick beachtet wird, muss nicht darauf geachtet werden, ob die Ausgaben ein zusammenhängendes Bild abgibt. Ein Beispiel für eine Technik für die Verschleierung einzelner Aufnahmen wird von Held [Hel22] vorgestellt. Das dort entwickelte „LokA“, berechnet eine Menge von Platzhalterobjekten, die den tatsächlichen Standort verschleiern sollen. Dabei sollen diese Objekte nicht offensichtlich sein. Man soll also nicht aufgrund der Wahrscheinlichkeit, dass der tatsächliche Standort an einer bestimmten Stelle ist, schließen können, welche Objekte nur Platzhalter sind. Außerdem wird auch darauf geachtet, dass die Wahrscheinlichkeit, mit der ein Platzhalterobjekt zu einem tatsächlichen Standort ausgegeben wird, keinen Einfluss auf seine Glaubwürdigkeit hat. Dazu werden mehrere Mengen von Platzhaltern gebildet und dann die ausgewählt, für die die Wahrscheinlichkeit, das die einzelnen Objekte erzeugt wurden, am ähnlichsten ist. Diese Technik kann ohne weitere Anpassungen in einem CV verwendet werden. Dafür muss lediglich eine verwendbare Implementierung vorhanden sein.

Im Gegensatz zu den Momentaufnahmen stehen die kontinuierlichen Aufnahmen. Würde man hier die gleichen Techniken verwenden wie zuvor, so wäre es leichter Manipulationen zu erkennen und sogar auch die ursprünglichen Daten zurückzuschließen. Dieses Problem wird zum Beispiel von Talukder und Ahamed [TA10] aufgefasst. Durch mehrere Anfragen (über einen oder mehrere

Nutzer) hat man mehr Informationen über den Standort, an dem sich jemand befindet und kann so Rückschlüsse auf den tatsächlichen Standort schließen. So lässt sich zum Beispiel der Bereich, in dem sich der tatsächliche Standort befinden einschränken. Dadurch kann es möglich sein, den zurückgelegten Weg mehr oder weniger genau zu bestimmen. Aus diesem Grund gibt es auch einige eigens für diese Anwendung entworfene Techniken. Zu diesen gehören zum Beispiel die Mix Zones, welche von Beresford und Stajano [BS03] für LBS definiert werden. Dort werden Zonen definiert, in welchen sich die Nutzer befinden. Für Nutzer, die in eine solche Zone eintreten, wird ein neuer bisher ungenutzter Bezeichner festgelegt. Tritt also ein Nutzer aus und verwendet dieses Pseudonym, kann nicht mehr erkannt werden, welcher der Nutzer, die in die Zone eingetreten sind, das ist. Sofern eine passende Implementierung vorhanden ist, kann diese Technik für CVs verwendet werden.

Methoden von Ortsdaten

Er wurde bereits eine große Anzahl an PETs für Ortsdaten entwickelt. Diese verwenden unterschiedliche Methoden und Berechnungsarten, um den Schutz der Privatsphäre zu verbessern. Daher ist es sinnvoll sie in mehrere Kategorien aufzuteilen. Die Aufteilung der Methoden wird in vier Kategorien vorgenommen. Zum einen gibt es die Methoden, die als *basierend auf Datenschutzrichtlinien (engl. privacy policy-based mechanisms)* bezeichnet werden. Hierbei ist das Verwalten der Richtlinien im Mittelpunkt. Die Hersteller des CV und Drittanbieter, welche in Abschnitt 2.2 erläutert wurden, sollen darin eingeschränkt werden, wie sie die Daten speichern, darauf zugreifen und nutzen. Dabei wird vom Anbieter einer Funktion erwartet, dass er sich an diese Richtlinien hält. Diese Kategorie ist hier nicht sonderlich relevant, da diese nicht vom Nutzer umgesetzt oder im in dieser Arbeit entworfenen System gespeichert werden kann. Eine solche Richtlinie kann wie beschrieben von Peterson [Pet05] beschreiben, wie ein Ortsobjekt aussehen soll. Innerhalb dieses Objektes ist festgelegt, wie und ob Daten weitergegeben werden dürfen. In dieser Kategorie befinden sich Techniken, bei denen der Anbieter einer Funktionalität bereit sein muss, die Richtlinien auch umzusetzen. Da das keine zentrale Softwarelösung darstellt, sondern auch die Initiative des Anbieters erfordern, werden diese Techniken für PriTeX nicht beachtet.

Die zweite Kategorie stellen die PETs dar, die *auf Verschleierung basieren (engl. obfuscation-based mechanisms)*. PETs in dieser Kategorie verändern den gesendeten Standort, im Vergleich zum tatsächlichen Standort, oder versuchen den zugehörigen Nutzer zu verstecken. Die Veränderung des Standorts kann dabei dadurch geschehen, dass statt des tatsächlichen Standortes immer ein leicht veränderter Standort gesendet wird. Es können aber auch mehrere andere Standorte gesendet werden, wobei dann nicht mehr erkennbar sein soll, wo sich die Person genau aufhält. Die Verschleierung des Nutzers geschieht durch das Zusammenspiel mit anderen Nutzern. Dabei können mehrere Nutzer zusammengefasst werden, wodurch keine Rückschlüsse auf den einzelnen möglich sein sollen. Es ist aber auch möglich, die Nutzer zu vermischen. Dabei werden die Bezeichner der Nutzer nach bestimmten Bedingungen vertauscht. Ein Beispiel für PETs in dieser Kategorie wird durch das Verhüllen des Pfades (engl. path cloaking) dargestellt. Eine Art davon wird von Hoh et al. [HGXA07] vorgestellt. Es soll nach der Verschleierung nicht mehr möglich sein, die genaue Route, von beispielsweise einem Fahrzeug, nachzuverfolgen. Das wird erreicht, indem man nur eine bestimmte Anzahl an Orten übermittelt, sodass ein Angreifer einen Nutzer nur für eine bestimmte Zeit verfolgen kann. Außerdem wird die Wahrscheinlichkeit, dass ein gesendeter Ort zu einem bestimmten Fahrzeug gehört, eingeschränkt.

Bei der dritten Kategorie handelt es sich um *kryptografiebasierte Mechanismen (engl. cryptography-based mechanisms)*. Diese zeichnen sich dadurch aus, dass sie die Privatsphäre mit der Hilfe von kryptografischen Methoden schützen. Das kann zum Beispiel durch Raumtransformation oder sichere Multi-party Computation (MPC). Insgesamt ist es das Ziel dieser Methoden, die privaten Inhalte unsichtbar für Angreifer zu machen. So auch bei der Technik von Ghinita et al. [GKK+08b]. Dort wird, mit der Hilfe von kryptografischen Methoden, verhindert, dass eine Anfrage auf einen bestimmten Nutzer zurückgeführt werden kann. Daher ist es nicht möglich nachzuverfolgen, was der Nutzer macht oder was seine Präferenzen sind. Das Szenario dieser Arbeit sieht vor, dass ein PET an einem zentralen Ort berechnet wird. Auch die Techniken, die eine Raumtransformation verwenden, benötigen die Kooperation des Anbieters. Daher sind kryptografiebasierte Methoden für PriTeX generell nicht für die Speicherung in PriTeX geeignet.

Die letzte Kategorie beschäftigt sich mit PETs *basierend auf Kooperation und Caching (engl. cooperation and chaching-based mechanisms)*. Hier wird versucht, die Anzahl der Aufrufe des LBS zu verringern. Dadurch soll das Risiko verringert werden, das entsteht, wenn sensible Daten weitergegeben werden. Dabei arbeiten mehrere Nutzer zusammen und speichern ihre Anfragen und die Antworten lokal ab. Dann kann ein anderer Nutzer die gespeicherten Daten verwenden, statt erneut eine Anfrage an den LBS zu senden. Das wird beispielsweise von Amini et al. [ALH+11] umgesetzt. Dort wird „Caché“ vorgestellt. Das System ermöglicht es, statt einem genauen Standort eine größere Region, an der der Nutzer interessiert ist, an den LBS gesendet. Dann werden die Daten auf dem eigenen Gerät gespeichert und Anfragen lokal bearbeitet. Befindet sich der Ort einer Anfrage nicht im Speicher, so können entweder keine relevanten Informationen gegeben werden oder der gewünschte Inhalt noch vom LBS angefordert werden. Da, wie in Abschnitt 2.2 beschrieben, ein CV zunächst mit dem Hersteller kommuniziert und keine direkte Kommunikation zwischen den CVs geschieht, sind Techniken, die das Verwenden, nicht für PriTeX geeignet. Aus diesem Grund werden im folgenden nicht die kooperationsbasierten Methoden, sondern nur die cachingbasierten Methoden beachtet.

Zusammenfassung

Nachdem zuvor die verschiedenen Kategorien der PETs für Ortsdaten beschrieben wurden, folgt nun eine Zusammenfassung der für das CV einsetzbaren Kategorien. Für die Ortsdaten wurden zwei Dimensionen von Kategorien identifiziert. Zum einen können die PETs für Ortsdaten nach ihrer Anwendungsart eingeteilt werden. Die Anwendungsart bezieht sich dabei auf die erwartete Anzahl der veröffentlichten Eingaben über die Zeit. Diese Dimension wird daher mit „Anzahl an Veröffentlichungen“ bezeichnet. Dabei gibt es zwei Kategorien, die sich nicht gegenseitig ausschließen. Sie wurden in Abschnitt 4.3.2 bereits erklärt. Die zwei Kategorien werden mit „Snapshot“ und „Kontinuierlich“ bezeichnet. Zum anderen können die PETs nach den Methodenarten eingeteilt werden. Dabei wird diese Dimension als „Kategorie“ bezeichnet. Die Kategorien, von denen in Abschnitt 4.3.2 festgestellt wurde, dass sie im hier angenommenen Szenario für das CV eingesetzt werden können, sind „Verschleierung“ und „Caching“.

In Tabelle 4.1 werden die zuvor im Text genannten Beispiele für PETs aus diesem Bereich gegeben. Dabei werden auch die hier nicht weiter beachteten PETs beschrieben. Um zu zeigen, welche Personengruppen ein PET in einem Fahrzeug schützen könnte, wenn es dort eingesetzt werden würde, werden auch die allgemeinen Kategorien der geschützten Personen, die bereits in Abschnitt 4.2.2 beschrieben wurden, genannt. Auch die für die Architektur ausschlaggebende

Name [Quelle]	Geschützte Personen	Anzahl an Datenquellen	Anzahl an Veröffentlichungen	Kategorie
PolicyFormat [Pet05]	Nutzer, Mitfahrer	Genau eine	Snapshot	Datenschutzrichtlinien
Lo*k*A [Hel22]	Nutzer, Mitfahrer	Genau eine	Snapshot	Verschleierung
Mix-Zones [BS03]	Nutzer, Mitfahrer	Mehr als eine	Kontinuierlich	Verschleierung
PathCloak [HGXA07]	Nutzer, Mitfahrer	Mehr als eine	Kontinuierlich	Verschleierung
PrivateRetrieval [GKK+08b]	Nutzer, Mitfahrer	Genau eine	Snapshot	Kryptographie
Caché [ALH+11]	Nutzer, Mitfahrer	Genau eine	Snapshot, Kontinuierlich	Caching

Tabelle 4.1: Tabelle zur Einteilung der PETs für Ortsdaten. Dabei sind die Namen teilweise aus den Quellen übernommen und teilweise selbst erdacht.

Anzahl der Datenquellen wird beschrieben. Der Einfluss der Anzahl der Datenquellen auf die verwendbare Architektur wurde bereits in Abschnitt 4.2.1 erläutert. Dadurch wird gezeigt, wie das Einfügen der PETs in die Kategorien erfolgen kann. Zum Beispiel kann das zuvor genannte „Caché“ [ALH+11] in diese Tabelle eingefügt werden. Es arbeitet immer auf dem Endgerät eines Nutzers, weswegen von genau einer Datenquelle eine Eingabe erwartet wird. Daher verwendet es „genau eine“ Datenquelle. Auch die Techniken aus [Pet05], [Hel22] und [GKK+08b] verwenden genau eine Datenquelle. Doch es gibt auch Techniken, die nicht nur eine Datenquelle verwenden und daher ihre Anzahl der Datenquellen mit „mehr als eine“ beschrieben wird. Hier wurden bereits zwei solcher Technike, vorgestellt, die in [BS03] und [HGXA07] beschrieben wird. Würde man die in Caché verwendete Methode leicht abändern und in einem CV einsetzen, so würde es die Ortsdaten des Fahrers und der Mitfahrer schützen. Generell werden durch PETs für Ortsdaten im Szenario des CV immer die Ortsdaten des Fahrers und der Mitfahrer geschützt, da die Ortsdaten des Fahrzeuges geschützt werden und sich der Fahrer und die Mitfahrer in diesem befinden. Caché ist darauf ausgelegt, den aktuellen Standort einmal zu versenden, um Informationen über nahegelegene interessante Orte zu speichern. Es erwartet daher eine einmalige Veröffentlichung, also einen „Snapshot“. Doch es kann auch für kontinuierliche Anfragen verwendet werden, da ein Bereich an Ortsdaten zwischengespeichert wird und dann eine Navigation durchgeführt werden kann. Daher kann auch die Anzahl der Veröffentlichungen auch „Kontinuierlich“ sein. Des Weiteren werden die gefundenen Orte zwischengespeichert um so nicht erneut den Standort senden zu müssen. Es basiert also auf „Caching“.

4.3.3 Stimmdaten

Viele moderne Autos beinhalten inzwischen einen PVA, auch Voice-controlled Digital Assistant (VCDA) genannt. Dieser kann verwendet werden, um Funktionen aufzurufen oder zu steuern, indem man mit dem Auto spricht. Das macht die Bedienung vieler Features einfacher und intuitiver. Allerdings wird hier eine große Menge an Daten aufgenommen und zur Verarbeitung weitergegeben. PVAs und Techniken für den Schutz der Privatsphäre lassen sich auf unterschiedliche Arten einteilen. Zum einen lassen sie sich durch ihre Aktivierungsart einteilen. Diese Einteilung ist von Vorteil, da sie direkten Einfluss auf die Nutzbarkeit des PETs für den Nutzer hat. Kommt der Nutzer mit einer bestimmten Art der Aktivierung nicht zurecht, so wird er das PET auch nicht verwenden. Das wird im nächsten Absatz erklärt. Im darauffolgenden Absatz erfolgt eine Beschreibung der Einteilung nach der Art der Verschleierungstechnik. Durch diese Einteilung kann die Anzahl der mögliche PETs auf eine überschaubare Anzahl an Gruppen reduziert werden und ein Grundverständnis der verwendeten Berechnungsweise der PETs einer Kategorie vermittelt werden.

Arten der Aktivierung

Die PVAs lassen sich laut Grey [Gre16] durch die Art ihrer Aktivierung in drei Gruppen einteilen. Zunächst gibt es die manuell aktivierten PVAs. Diese werden mit einem physischen oder digitalen Knopf in den „zuhören“ Modus versetzt. Das hat den Vorteil, dass ein PVA nicht ungewollt Daten aufnimmt und weitergibt. Man hat dadurch aber immer einen Aufwand, um das Gerät wieder einzuschalten und es zu benutzen, was mit den anderen Arten nicht nötig ist. Die zweite Art sind Stimmaktivierte PVAs. Diese hören immer zu, zeichnen aber nur Daten auf, nachdem ein vorherbestimmtes Wort, das *Wake Word*, gehört haben. Der Vorteil dieser Aktivierungsart ist, dass Nutzer nicht manuell auf einen Knopf drücken müssen, um den PVA zu aktivieren, sondern einfach ein bestimmtes Wort sagen und den PVA verwenden können. Durch diese Art der Aktivierung ist die Privatsphäre geschützt, solange das vorbestimmte Wort nicht gesagt wird. Es kommt allerdings zu Problemen, wenn ähnliche Worte gesagt werden. Als Letztes gibt es die PVAs, die immer an sind. Das ist nützlich, wenn man zum Beispiel ein Überwachungssystem aufbauen möchte. Da das Gerät immer zuhört und die gehörten Daten verarbeitet, kann es Geräusche, die in Verbindung mit Einbrüchen stehen, wie Glassplitter oder dem Brechen von Holz, hören und darauf reagieren.

Je nachdem welche Art von PVAs benutzt werden, müssen unterschiedliche Techniken für den Schutz der Privatsphäre verwendet werden. So ist es leicht, private Gespräche zu verstecken, wenn man einen manuell aktivierten PVA benutzt. Diese Aufgabe wird allerdings deutlich schwerer, wenn man ein mikrofongesteuertes Gerät betrachtet, das immer an ist. Denn auch die PETs, die für den Schutz von Stimmdaten bestimmt sind, können auf diese drei Arten aktiviert werden. Dabei sollte die Aktivierungsart des PET an die Aktivierungsart des Geräts und die eigenen Ansprüche an die Privatsphäre anpassen. So ist beispielsweise ein PET, das immer an ist, bei einem PVA der manuell aktiviert wird nicht nötig, da auch nicht aus Versehen Daten aufgezeichnet werden. Da man für die Aktivierung des PVAs bereits einen Knopf umlegen muss, könnte man das PET zusammen mit dem PVA, also mit dem gleichen Knopf, aktivieren und deaktivieren.

Cheng et al. [CBYR18] stellen ein PET vor, welches sich durch Wake Word aktiviert. Dazu wird ein sogenanntes *Protection Jamming Device (PJD)* eingesetzt, welches auf gesprochene Wake Words, die zu PVAs gehören, achtet. Es muss diese schneller erkennen als der zugehörige PVA, damit er ein akustisches Signal aussenden kann, welches es für den PVA unmöglich macht das gesprochene

zu verstehen. Nur durch die schnelle Erkennung kann sichergestellt werden, dass der PVA das Wake Word nicht hört. Das Ziel ist es, dass eine Unterhaltung nicht aufgezeichnet wird, nachdem ein Wake Word gesagt wurde. Da ein PJD verwendet wird, welches ein eigenes Gerät ist, benötigt diese Technik eigene Hardware. Außerdem wird mit dieser Technik jegliche Funktionalität unterbunden. Sie ist also für die Speicherung in PriTeX nicht geeignet.

Arten von Verschleierungstechniken

Die Angriffsflächen und die Verschleierungstechniken für Stimmdaten kann in verschiedene Arten eingeteilt werden. Cheng und Roedig [CR22] beschreiben vier verschiedene Arten. Diese werden im Folgenden erläutert und Beispiele werden genannt. Interessant ist dabei, dass sowohl Software- als auch Hardwarelösungen entwickelt wurden. Zunächst werden daher die Arten an Verschleierungtechniken beschrieben, die einen Schutz der Privatsphäre mit der Hilfe von Hardware ermöglichen. Danach werden die Softwarelösungen beschrieben.

Als Erstes gibt es *die akustischen Denial of Service (DoS) Techniken*. Grundlegend für diese Techniken ist die Idee, den PVA mit einem akustischen Signal so zu stören, dass er keine anderen Daten aufnehmen kann, beziehungsweise die Aufnahmen nicht nutzbar sind. Dadurch können aus den Aufnahmen keine privaten Daten mehr abgeleitet werden und somit ist die Privatsphäre geschützt. Dabei gibt es akustische Signale, die innerhalb des hörbaren Bereichs liegen. Da diese aber für den Nutzer störend sind, kann auch ein Signal verwendet werden, das außerhalb des für Menschen Hörbaren Bereichs liegt. Dadurch wird ein angenehmes Nutzererlebnis geschafften. Für diese Art der Techniken haben Gao et al. [GCFB18] ein Gerät entwickelt, welches immer ein akustisches Signal aussendet, sodass PVAs in der Nähe die Unterhaltungen nicht mithören können. Dieses Blockieren wird erst aufgehoben, wenn der Nutzer ein akustisches Signal gibt. Dabei liegt das akustische Signal im Ultraschallbereich. Außerdem wird auch eine Authentifizierung durchgeführt, die verhindert, dass jemand ohne die Berechtigung das Gerät abschalten kann. Die Techniken aus diesem Bereich verhindern die Aufzeichnung von Stimmdaten komplett. Es wird davon ausgegangen, dass schon dem Mikrofon beziehungsweise dem Endgerät nicht vertraut werden kann.

Nachdem nun die Art der Verschleierungtechniken beschrieben wurde, die den Schutz mit der Hilfe von Hardware schützen, wird klar, dass diese Techniken nicht für die Speicherung in PriTeX geeignet sind. Das liegt daran, dass hier, wie zuvor beschrieben, davon ausgegangen wird, dass das Endgerät nicht vertrauenswürdig ist. Dadurch, dass in dieser Arbeit das CV allerdings als vertrauenswürdig angenommen wird, wird auch das Mikrofon als vertrauenswürdig angesehen. Zum anderen wurde bereits in Kapitel 3 beschrieben, dass PriTeX keine Hardwarelösungen speichern kann. Insgesamt sind diese Techniken für PriTeX nicht relevant sind. Im Folgenden werden nun die Softwarelösungen vorgestellt.

Die zweite Art der Techniken beschäftigt sich mit der *Zugriffskontrolle (engl. access control)*. Hier besteht die Herausforderung darin, sicherzustellen, dass niemand Zugriff auf die Funktionen und gespeicherten Daten bekommt, der nicht dafür zugelassen ist. Ein Problem dabei ist, dass es meist einen Besitzer und mehrere Nutzer gibt. Dann hat meist nur eine Person Zugriff auf die gesammelten Daten und die Kontrolle über diese. Dadurch kann eine Person im Haushalt, welche Zugriff auf den PVA hat, die Kontrolle über personenbezogene Daten und Geheimnisse der anderen, im Haushalt lebenden, Personen. Diese Problematik kann auch auf den Fahrer und die

Mitfahrer eines CVs übertragen werden. Erstellt ein PVA automatisch neue Nutzer, wenn er eine Stimme nicht erkennt, so hat der Besitzer des PVA auch Zugriff auf diese Aufnahmen, also auf Aufnahmen von nebenstehenden Personen. Außerdem kann ohne Zugriffskontrolle jeder einen PVA benutzen, solange er in der Reichweite eines solchen ist. Dadurch können zum Beispiel Bestellungen aufgegeben werden, wenn der PVA mit einem entsprechenden Dienst verbunden ist. Ein Beispiel für ein PET, welches den Zugriff auf ein Gerät einschränkt, wird von Zhang et al. [ZTYC16] vorgestellt. Dort wurde eine Technik entwickelt, mit der es möglich ist, eine tatsächlich gesprochene Stimme von einer Aufnahme der gleichen Stimme zu unterscheiden. Dabei werden die Eigenschaften des Stimmapparats ausgenutzt. Durch seine Form dauert es für unterschiedliche Laute unterschiedlich lange zu den Stereomikrofonen eines Smartphones zu gelangen. Da eine von einem Lautsprecher ausgegebene Aufnahme diese Eigenschaften nicht aufweist, ist es nicht möglich, mit dieser auf das Gerät Zugriff zu bekommen. Für den Einsatz in CVs müsste diese Technik angepasst werde, da die Mikrofone in einem Auto anders platziert sind, als bei einem Smartphone. Die Techniken, die Zugriffskontrolle umsetzen, lösen zunächst ein Sicherheitsproblem, bei welchem der Zugriff auf ein Gerät, also auch ein Auto, kontrolliert wird. Doch der Zugriff auf ein Gerät bringt auch eine Gefahr für die Privatsphäre mit sich. Denn wer Zugriff auf ein Gerät hat, der hat auch Zugriff auf die darauf gespeicherten Daten und kann so private Daten einsehen. Daher sind Techniken aus diesem Bereich auch für die Privatsphäre relevant und müssen berücksichtigt werden.

Die dritte Art der Angriffsflächen und Verschleierungstechniken beschäftigt sich mit der *Voice Privacy*, also mit der Privatsphäre der Stimme. Dieser Bereich beschäftigt sich mit den personenbezogenen Daten, die aus den Stimmdaten abgeleitet werden können. So kann man eine Person allein anhand ihrer Sprechweise und Stimme eindeutig zu identifizieren. Dadurch können Stimmdaten und damit auch einzelne Aufnahmen auf eine Person zurückgeführt werden. Mit dem heutigen Stand der Technik ist es außerdem möglich, Emotionen aus Stimmdaten abzuleiten. Da Emotionen jedoch sehr privat sind, wird daran gearbeitet, dass PVAs nicht mehr in der Lage sind, diese aus den Daten zu lesen. Noch gefährlicher ist die Tatsache, dass man über Stimmdaten auch bestimmte Einzelheiten über den gesundheitlichen Zustand herausfinden kann. Diese Daten gelten als besonders schützenswert. Nelus und Martin [NM19] stellen ein Schema vor, mit welchem es möglich ist, die Stimme so zu verschleiern, dass sie nicht mehr der Person zugeordnet werden kann. Dazu kommen drei Deep Neural Networks zum Einsatz. Das erste soll die identifizierenden Eigenschaften der Stimme minimieren. Das zweite soll danach herausfinden, ob der Sprecher männlich oder weiblich ist. Gleichzeitig soll das dritte versuchen den, Sprecher zu identifizieren. Dabei stellen sie fest, dass bei der Steigerung der extrahierten Eigenschaften das zweite Deep Neural Network nicht wesentlich inakkurater wird, während das dritte sehr inakkurat wird. Damit soll gezeigt werden, dass die Privatsphäre nicht immer für das Umsetzen von Funktionen aufgegeben werden muss. Diese Technik kann, sofern die entsprechende Rechenleistung und Implementierung vorhanden ist, auch in einem CV eingesetzt werden.

Die letzte beschriebene Art der Verschleierungstechniken behandelt das akustische Abtasten (engl. acoustic sensing). Diese Art der PETs, die vor einer Abtastung der Umgebung mit der Hilfe von akustischen Signalen. Dazu werden die akustischen Signale und Reflexionen in der Umgebung verwendet, um Rückschlüsse auf die Vorgänge in der Nähe des Mikrofons oder der Mikrofone, zu erhalten. Dadurch können beispielsweise die Muster der Atmung erkannt werden, wie es in der Arbeit von Xu et al. [XYC+19] gemacht wird. Diese Technik verwendet maschinelles Lernen für das Herausfiltern der Atemmuster. Dabei werden die Atemmuster des Fahrers eines Fahrzeuges

während der Fahrt analysiert. Für den Schutz vor akustischer Abtastung fanden Cheng und Roedig [CR22] keine PETs und auch eine Literatursuche fand keine Ergebnisse. Daher wird diese Art von Verschleierungstechniken im folgenden nicht weiter behandelt.

Zusammenfassung

Im Folgenden wird eine Zusammenfassung der zuvor beschriebenen PETs gegeben. Dabei wurden zuvor zwei Dimensionen für die Kategorisierung beschrieben. Die erste Dimension beschreibt, wie ein PET aktiviert wird, weswegen die auch als „Aktivierungsart“ bezeichnet wird. Dort sind drei Kategorien möglich. Diese sind „Manuell“, „Wake Word“ und „Immer an“. Die zweite Dimension beschreibt die Arten der Verschleierungstechniken. Sie wird als „Kategorie“ bezeichnet. Zuvor wurde für zwei Kategorien festgestellt, dass sie im hier angenommenen Szenario passend sind. Das ist zum einen die „Zugriffskontrolle“ und zum anderen die „Privatheit der Stimme“.

Name [Quelle]	Geschützte Personen	Anzahl an Datenquellen	Aktivierungsart	Kategorie	Hardware
JPVA [CBYR18]	Nutzer, Mitfahrer, Beistehender	Genau eine, mehr als eine	Wake Word	Denial of Service	Ja
VoiceLive [ZTYC16]	Nutzer, Mitfahrer	Genau eine	Immer an	Zugriffskontrolle	Nein
JA [GCFB18]	Nutzer, Mitfahrer, Beistehender	Genau eine, mehr als eine	Immer an	Denial of Service, Zugriffskontrolle	Ja
KIG [NM19]	Nutzer, Mitfahrer, Beistehender	Genau eine	Immer an	Privatheit der Stimme	Nein

Tabelle 4.2: Tabelle zur Einteilung der PETs für Stimmdaten. Dabei sind die Namen teilweise aus den Quellen übernommen und teilweise selbst erdacht.

Eine Übersicht und die im Text genannten Beispiele, für mögliche Beschreibungen von PETs, mit den Dimensionen der Kategorien können in Tabelle 4.2 angesehen werden. Dabei werden auch die Beispiele aus den nicht für das angenommene Szenario passenden Kategorien beschrieben. Zusätzlich wird genannt, welche der Techniken Hardware in ihrer Lösung verwenden. Diese sind, wie zuvor bereits beschrieben, nicht für PriTeX geeignet. Wie schon in der Zusammenfassung der PETs für die Ortsdaten, Abschnitt 4.3.2 wird auch hier auf die geschützten Personen und die Anzahl an Datenquellen eingegangen. In der Tabelle wird zum Beispiel die Technik von Nelus und Martin [NM19] beschrieben. Diese Technik kann, wenn sie auf die Daten aus einem CV eingesetzt wird, die Stimme des Fahrers oder „Nutzers“ der „Mitfahrer“ und der „Beistehenden“ verschleiern. Dabei wird davon ausgegangen, dass ein Fahrzeug sowohl nach innen als auch nach außen gerichtete Mikrofone besitzt. Auch die Techniken aus [CBYR18] und [GCFB18] schützen die Privatsphäre von Fahrer, Mitfahrern und Beistehenden geschützt werden. Die in [ZTYC16] angegebene Technik schützt hingegen die Privatsphäre des Fahrers und der Mitfahrer, da sie sich auf Zugriff auf die

innen liegenden Mikrofone hat, über welche die Funktionen innerhalb des Fahrzeuges gesteuert werden können. Da sich der Beistehende nicht im Fahrzeug befindet, kann diese Technik ihn deshalb auch nicht schützen. Dabei werden in den hier vorgestellten Techniken aus [ZTYC16] und [NM19] nur die Daten von einem CV, also einem Endgerät oder CV verwendet. Es gibt also „genau eine“ Datenquelle. Die Techniken aus [CBYR18] und [GCFB18] können sowohl für eine, als auch mehrere Datenquellen verwendet werde. Das liegt daran, dass der Schutz mithilfe von Hardware aufgebaut wird. Dadurch kann sich ein Beistehender vor den Mikrofonen mehrerer CVs schützen. Aber es ist auch möglich, sich nur vor einem CV zu schützen. Der Schutz ist also unabhängig von der Anzahl der Datenquellen. Da in der in [NM19] beschrieben Technik alle Daten so verschleiert werden, ist dieses PET „immer an“. Außerdem versteckt es, wie beschrieben, den Sprecher und ist daher in der Kategorie „Privatheit der Stimme“. Zuletzt ist anzumerken, dass keine spezielle Hardware für die Nutzung dieser Technik nötig ist. Auch für die in [ZTYC16] beschriebene Technik ist keine Hardware nötig. Im Gegensatz dazu benötigen die Techniken aus [CBYR18] und [GCFB18] Hardware.

4.3.4 Kameradaten

Kameradaten sind ein Datentyp, aus dem sich viele unterschiedliche sensible Daten ableiten lassen. So lässt sich aus Kameradaten beispielsweise herauslesen, wo sich ein Fahrzeug oder eine Person aufgehalten hat und was die Route war. Das wird von Schindler et al. [SBS07] getan. Doch es ist auch möglich noch viele weitere sensible Informationen aus den Kameradaten herauslesen. Orekondy et al. [OSF17] definieren einige dieser Attribute, wie Geschlecht, Tattoos und viele andere. Außerdem untersuchten sie, wie wichtig diese Daten für Nutzer sind und wie gut diese darin sind, ihre sensiblen Daten verstecken können. Aus diesen Gründen ist es wichtig, die sensiblen Informationen in den Kameradaten zu schützen. Dazu wurden bereits viele PETs entworfen. Im Folgenden soll für diese eine Einteilung in Kategorien vorgenommen werden.

Anzahl der Aufzeichnungen über die Zeit

Wie bei den Ortsdaten lassen sich auch hier die Techniken nach der Anzahl der Aufzeichnungen über die Zeit einteilen. Dabei lassen sich erneut zwei Kategorien erkennen. Da ist zum einen das Bild oder die Einzelaufnahme oder der Snapshot und zum anderen die Folge von Bildern oder das Video, also die kontinuierliche Aufnahme. Oft können zwar ähnliche Techniken verwendet werden, aber dennoch ergeben sich, je nachdem in für welche Kategorie das PET entworfen wurde, unterschiedliche Herausforderungen. Ein Beispiel für Techniken, für die diese Unterscheidung wichtig ist, stellen die Methoden zur Objektentfernung dar. Sie werden von Padilla-López et al. [PCF15] beschrieben. Dabei sollen Personen und Objekte aus einem Bild oder Video entfernt werden, um so sensible Daten zu schützen. Um die Lücke zu füllen, können verschiedene Techniken verwendet werden. So beispielsweise die Technik von Bertalmio et al. [BSCB00]. Dort können Bereiche eines Bildes ausgewählt werden und werden dann so ersetzt, dass ein zusammenhängendes Bild entsteht. Sie konzentrieren sich dabei auf das Ersetzen des Bereiches und nicht das Auswählen dieses Bereichs. Dazu werden die vorhandenen Kanten außerhalb des zu ersetzenden Bereichs verwendet und innerhalb dieses Bereichs weitergezogen. Dabei muss der Nutzer der Technik nicht festlegen, mit welchen Informationen der Bereich gefüllt werden soll und die Technik kann den Bereich automatisch füllen.

Für Videos können jedoch nicht die gleichen Techniken verwendet werden. Bei den PETs für Videos ist es wichtig auch auf den Zusammenhang zwischen den Bildern oder Rücksicht zu nehmen. So kann aus einem Video möglicherweise mehr über den Ort der Aufnahme herausgefunden werden, da es nicht nur eine Aufnahme ist, sondern mehr vom Hintergrund gezeigt wird. Auch ist eine exakte Erkennung der sensiblen Bereiche von größerer Wichtigkeit bei einem Video. Denn wird der sensible Bereich nicht genau erkannt und in unterschiedlichen Einzelbildern der Videos werden unterschiedliche Teile dieses Bereichs verdeckt, so kann mehr über den Bereich in Erfahrung gebracht werden, als bei einem einzelnen Bild mit der gleichen Fehlerrate der Erkennung. Diese Unterschiede müssen von PETs für Kameradaten berücksichtigt werden, um eine tatsächlich sichere Technik zu entwickeln. Für die Objektentfernung kann dabei beispielsweise die Technik von Zhang et al. [ZXS05] verwendet werden. Sie verwenden eine Bewegungsschicht, um die zeitliche Kohärenz zwischen den Einzelbildern zu erhalten. Für das Entfernen von Objekten werden zunächst synthetische Schichten erzeugt, aus welchen anschließend ein neues und plausibles Video erzeugt wird. Im Gegensatz zu den Techniken für die Objektentfernung in Bildern nutzt diese Technik also die zeitlichen Informationen des Videos, um ein plausibles Ergebnis zu erhalten.

Kategorien von Methoden

Um sensible Daten in Kameradaten zu schützen, gibt es PETs aus verschiedenen Kategorien. Solche Kategorien werden beispielsweise von Padilla-López et al. [PCF15] eingeführt. Sie benennen dabei fünf Kategorien, welche im Folgenden genannt und beschrieben werden.

Die erste Kategorie ist die *Intervention (engl. intervention)*. PETs in dieser Kategorie setzen direkt an der Quelle des Datensammelns an. Sie sollen verhindern, dass eine Kamera überhaupt Daten aufzeichnet. Dabei geht es nicht darum, alle Aufzeichnungen zu verhindern, sondern das Aufzeichnen in bestimmten Bereichen und von bestimmten Objekten unmöglich zu machen. Dabei gibt es zwei Ansätze. Der erste Ansatz verhindert die Aufnahme von Bildern und Videos. Dazu werden Techniken eingesetzt, die das Bild der Kamera stören. So gibt es beispielsweise ein PET, welches Kameras aufspüren kann und diese dann mit blinkendem Licht verwirren kann. Dieses wurde von Patel et al. [PST09] entworfen und funktioniert für bestimmte Kameratypen, wie zuvor beschrieben. Der zweite Ansatz setzt auf der Seite der Software an. Dazu wird auf Kameras oder Geräte mit Kamera eine Firmware eingespielt, die noch vor der Speicherung der Daten eingreifen kann. So zum Beispiel in der Arbeit von Winkler et al. [WER14]. Diese entwickelten eine Architektur für Kameras, in welcher eine Einheit zum Schutz der Privatsphäre vor ein typisches Kamerasystem geschaltet wird. Diese Einheit legt einen Cartoon Effekt auf das gesamte Bild, wodurch die Privatsphäre geschützt wird. Techniken aus dieser Kategorie verwenden meistens eine zusätzliche Hardware, weswegen sie in PriTeX nicht gespeichert werden können. Des Weiteren wird hier bereits die Aufzeichnung von Kameradaten verhindert, wobei davon ausgegangen wird, dass der Hardware und dem Endgerät nicht vertraut wird. Da hier aber von einem vertrauenswürdigen CV und somit einer vertrauenswürdigen Kamera ausgegangen wird, sind diese Techniken für PriTeX nicht relevant.

In der zweiten Kategorie geht es um das *blinde Sehen (engl. blind vision)*. Diese und die folgenden Kategorien setzen alle nach der Aufnahme der Daten an. Mit blindem Sehen ist gemeint, dass die Parteien, die mit den Daten Berechnungen ausführen, diese nicht sehen können. Das wird durch sicheres MPC erreicht. Dabei soll die Interaktion wie folgt aussehen. Die Partei mit den Daten kann den Algorithmus der berechnenden Partei nutzen, ohne dass sie etwas über den Algorithmus erfährt.

Gleichzeitig soll die berechnende Partei aber auch nichts über die Daten erfahren. So ein PET wird von Shashanka [Sha10] entworfen. Dabei wird sicheres MPC verwendet, um Gaussian Mixture Model Berechnungen sicher für die Privatsphäre umzusetzen. Die Daten sind dabei auf mehr als einen Nutzer verteilt. Techniken aus dieser Kategorie verlangen also ein Zusammenspiel zwischen dem Endgerät des Nutzers und einer Cloud. Da wir jedoch, wie in Abschnitt 4.2.1 beschrieben, davon ausgehen, dass die Berechnung immer nur entweder auf dem Fahrzeug oder bei einer dritten Partei ausgeführt wird, sind diese Techniken für PriTeX nicht geeignet.

Die dritte Kategorie befasst sich mit der *sicheren Verarbeitung (engl. secure processing*). Hierunter fallen alle PETs, die nicht mit sicherem MPC arbeiten, die Daten aber dennoch auf eine Art und Weise verarbeiten, die die Privatsphäre der Nutzer respektiert. Zhang et al. [ZTC12] haben eine solche Technik entworfen. Sie beschrieben, wie man mithilfe der Tiefeninformationen die Aktivitäten von aufgezeichneten Personen verfolgen kann. Diese Personen sollen Senioren sein und es soll möglich sein, einen Fall zu erkennen. Mit den Tiefeninformationen kann erkannt werden, was die Aktivität der beobachteten Person ist. Die RGB Werte werden nur dann angesehen, wenn die Person außerhalb des Bereichs zur Aufzeichnung der Tiefeninformationen ist. Selbst wenn die RGB Werte verwendet werden, wird der Hintergrund aus dem Bild abgezogen und nur die Vordergrundmasken, welche nur Histogramme der Breite-Höhe Verhältnis enthält, für die Visualisierung verwendet. Dadurch soll die Privatsphäre geschützt werden. Die PETs dieser Kategorie werden in PriTeX nicht berücksichtigt. Das liegt daran, dass hier die keine Berechnung auf dem Endgerät des Nutzers stattfindet. Stattdessen müssen die Techniken aus dieser Kategorie vom Anbieter einer Funktion umgesetzt werde.

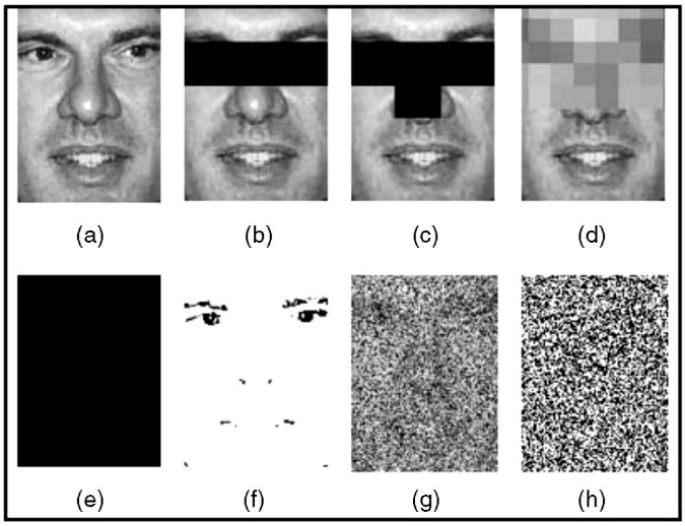


Abbildung 4.3: Beispiel für Deidentifizierungsmethoden, vorgestellt von [NSM05].
(a) Originalbild, (b) Balkenmaske, (c) T Maske, (d) Pixelierung, (e) Schwärzen, (f) Schwelle, (g) Zufällige Grauwerte, (h) Zufällig in Schwarz und Weiß.

Das *Redigieren oder Überarbeiten (engl. redaction)* von sensiblen Bereichen, in einem Bild oder Video, umfasst die vierte Kategorie. Dabei wird in zwei Schritten gearbeitet. Der erste Schritt ist das Erkennen von sensiblen Bereichen. Sind diese einmal erkannt, kann eine Vielzahl möglicher Überarbeitungen vorgenommen werden. Beispielsweise kann man die Bereiche mit einer schwarzen Form überschreiben. Aber auch verschiedene Arten von Verwischung oder Deidentifizierung können durchgeführt werden. Auch das ersetzten der Bereiche durch abstrahierte Objekte der gleichen Art

ist möglich. Einige dieser Möglichkeiten werden in Abbildung 4.3 bildlich dargestellt. Zum Beispiel zeigt Abbildung 4.3(b) die Verschleierung mit einer Balkenmaske über den Augen. Dabei werden die Augen einer Person mit einem schwarzen Balken überdeckt. Von Newton et al. [NSM05] wurde ein Algorithmus entworfen, mit dem man Gesichter verschleiern kann. Dieser wird k-Same genannt. Das Ziel ist es, dass Face Recognition Software ein Gesicht nicht mit Sicherheit identifizieren kann. Dazu werden die k ähnlichsten Gesichter betrachtet. Dann wird ein neues Gesicht aus dem Durchschnitt der Gesichter erstellt. Dieses neue Gesicht wird auf die k ursprünglichen Gesichter gesetzt. Jetzt gibt es also k Personen mit dem gleichen Gesicht, wodurch sie nicht mehr eindeutig erkennbar sind. Da es sich bei den PETs dieser Kategorie um Software Lösungen handelt, ist es möglich diese PETs in PriTeX einzufügen. Diese Technik könnte beispielsweise für die Aufnahmen der Außenkameras eines CVs eingesetzt werden. Viele dieser PETs können mit leichten Anpassungen im CV eingesetzt werden. Dabei können die Techniken übertragen werden, die einen ähnlichen Anwendungsfall haben. So kann beispielsweise ein PETs, welches die Informationen eines Gesichts schützen kann, auch im Fahrzeug für die gleiche Anwendung verwendet werden. Dennoch ist es nötig auch Techniken zu entwickeln, die für das Szenario des CV ausgerichtet sind. Denn dort steht für die inneren Sensoren der Fahrer im Mittelpunkt, um beispielsweise Sekundenschlaf zu erkennen. Dafür ist es wichtig, Techniken zu entwickeln, die diese Funktionen weiter zulassen. Des Weiteren sind Techniken aus diesem Bereich Softwarelösungen. All das macht sie für PriTeX relevant.

Die fünfte und letzte Kategorie beschäftigt sich mit dem *Verstecken von Informationen (engl. information hiding)*. Wie auch beim Redigieren oder Überarbeiten, werden hier zunächst sensible Bereiche erkannt. Danach werden diese jedoch nicht nur verdeckt oder überschrieben, sondern man versteckt sie. Das bedeutet, dass die Informationen nicht gelöscht werden, sondern lediglich so verändern werden, dass sie mit der richtigen Methode in der Zukunft wieder sichtbar werden. Dazu kann man, wie im Vorschlag von Paruchuri und Cheung [PC08], die Informationen in ausgewählte Koeffizienten der diskreten Kosinustransformation einbetten. Dabei werden die Koeffizienten so ausgesucht, dass eine Kostenfunktion minimiert wird. Diese beschreibt der Zusammenhang der Verzerrung und Bitrate, wobei die Gewichtung vom Nutzer gewählt werden kann. Auch die PETs dieser Kategorie sind Software Lösungen und können somit in PriTeX gespeichert werden.

Zusammenfassung

Dieser Abschnitt soll die Feststellungen aus den vorherigen Abschnitten zusammenzufassen. Dabei wurden zuvor zwei Dimensionen von Kategorien erläutert. Zum einen wird die Dimension genannt, welche die Anzahl der Veröffentlichungen oder Aufzeichnungen über die Zeit, die von einem PET als Eingabe erwartet werden, beschreibt. Sie wird mit „Anzahl der Veröffentlichungen“ bezeichnet. PETs können in dieser Dimension zu zwei Kategorien gehören, wobei es möglich ist auch zu beiden gleichzeitig zu gehören. Die beiden Kategorien werden als „Snapshot“ und „Kontinuierlich“ bezeichnet. Zum anderen gibt es die Dimension der Kategorien von Methoden, welche als „Kategorie“ bezeichnet wird. Die zum angenommenen Szenario passenden Kategorien sind dabei „Redigieren“ und „Verstecken“.

Eine Übersicht über die Einteilungen einiger zuvor bereits genannten beispielhafter PETs wird in Tabelle 4.3 gegeben. Dabei werden auch die Beispiele genannt, die nicht zu einer der passenden Kategorien gehören. Des Weiteren wird gezeigt, ob eine Technik eine spezielle Hardware verwendet. Dadurch ist sie für eine Speicherung in PriTeX nicht geeignet. Auch in dieser Tabelle werden wieder, wie in Abschnitt 4.3.2 und Abschnitt 4.3.3, die geschützten Personen und die Anzahl an

Name [Quelle]	Geschützte Personen	Anzahl an Datenquellen	Anzahl an Veröffentlichungen	Kategorie	Hardware
BlindSpot [PST09]	Nutzer, Mitfahrer, Beistehender	Genau eine, mehr als eine	Snapshot, Kontinuierlich	Intervention	Ja
TrustEYE.M4 [WER14]	Nutzer, Mitfahrer, Beistehender	Genau eine, mehr als eine	Snapshot, Kontinuierlich	Intervention	Ja
GMMPC [Sha10]	Nutzer, Mitfahrer, Beistehender	Mehr als eine	Snapshot	Blindes Sehen	Nein
RGBFalling [ZTC12]	Beistehender	Genau eine	Kontinuierlich	Sichere Verarbeitung	Nein
FDI [NSM05]	Nutzer, Mitfahrer, Beistehender	Genau eine	Snapshot	Redigieren	Nein
DCTHiding [PC08]	Fahrer, Mitfahrer, Beistehender	Genau eine	Kontinuierlich	Verstecken von Informationen	Nein

Tabelle 4.3: Tabelle zur Einteilung der PETs für Kameradaten. Dabei sind die Namen teilweise aus den Quellen übernommen und teilweise selbst erdacht.

Datenquellen genannt. Eines der verwendeten Beispiele ist von Zhang et al. [ZTC12]. In diesem wird eine Technik für die Fallerkennung eingesetzt. Diese Erkennung ist nur für „Beistehende“ und nicht für den Fahrer oder die Beifahrer von Bedeutung. Im Gegensatz dazu können die anderen beschriebenen Techniken für alle Personengruppen verwendet werden. Dazu müssen sie je nach Anwendungsfall angepasst werden. Auch werden in [ZTC12] die Daten von „genau einer“ Datenquelle verwendet. Diese Datenquelle sendet wiederum kontinuierlich Daten. Da dabei, in bestimmten Fällen, nur auf den Tiefeninformationen gerechnet wird, ist sie in der Kategorie der „sicheren Verarbeitung“. Sie benötigt keine spezielle Hardware, die nur für diese Technik verwendet wird, um den Schutz zu gewährleisten. Andere Techniken, die ebenfalls nur eine Datenquelle verwenden, werden in [NSM05] und [PC08] vorgestellt. Einige Techniken, die im Gegensatz dazu mehr als eine Datenquelle verwenden, werden in [Sha10] und [NSM05] beschrieben. Außerdem werden in [PST09] und [WER14] Techniken vorgestellt, die sowohl für einzelne als auch für mehrere Datenquellen verwendet werden können. Das liegt daran, dass sie eine eigene Hardware verwenden, die außerhalb des CVs verwendet werden kann und so von der genauen Anzahl der Datenquellen unabhängig ist. Des Weiteren verwenden [PST09] und [WER14] Hardware, um den Schutz zu gewährleisten. Die anderen Techniken verwenden keine Hardware. Die Anzahl an Veröffentlichungen, die erwartet werden, spielt für einige Techniken keine Rolle. Diese Techniken

werden in [PST09] und [WER14] beschrieben. Die in [NSM05] und [Sha10] beschriebenen Techniken kann Snapshots verschleiern. Im Gegensatz dazu können die in [ZTC12] und [PC08] beschriebenen Techniken sensible Daten in kontinuierliche Veröffentlichungen schützen.

4.4 Metadaten der Implementierung

Nachdem zuvor die Metadatenfelder der PETs identifiziert wurden, sollen im folgenden noch die nötigen Metadatenfelder der Implementierungen dieser PETs identifiziert werden. Zuvor wurde in Abschnitt 4.1 bereits beschrieben, sollen auch die Implementierungen die Anforderung der Adressierbarkeit (A2) erfüllen. Dazu muss eine Implementierung zunächst identifizierbar sein.

Identifikator. Wie auch die PETs, sollen auch ihre Implementierungen eindeutig identifizierbar sein. Dazu kann dieses Metadatenfeld verwendet werde. Hier kann ein eindeutiger Identifikator oder kurz eine ID für eine Implementierung eindeutig gewählt werden.

Version. Da viele Programme in verschiedenen Version herausgebracht werden, um sie kontinuierlich zu verbessern, muss für die Adressierbarkeit (A2) auch diese Version dokumentiert werden. Dazu ist dieses Metadatenmodell geeignet.

Referenz. Um eine Implementierung herunterladen zu können, muss eine Referenz auf den Speicherort gegeben werden. Diese Referenz kann in diesem Metadatenfeld gespeichert werden.

Um die weiteren Anforderungen der Vertrauenswürdigkeit (A4) und der Sicherheit (A5) Umzusetzen müssen noch weitere Metadatenfelder eingefügt werden. Dazu werden die folgenden Metadatenfelder verwendet.

Hash. Der Hash ist dazu da, dass ein Nutzer überprüfen kann, ob die Implementierung geändert wurde oder der Implementierung entspricht, die im System beschrieben wird. Die Grundlagen der kryptografischen Hashfunktionen werden von Sobti und Geetha [SG12] beschrieben. Allgemein weisen Hashfunktionen einer beliebig langen Eingabe einen String mit fester Länge zu. Für die gleiche Eingabe wird auch der gleiche Hash ausgegeben. So kann überprüft werden, ob die Datei einer Implementierung der in PriTeX gespeicherten Version entspricht. Durch dieses Metadatenfeld kann also die Vertrauenswürdigkeit (A4) der Implementierung überprüft werden.

Zertifikat. Wie zuvor in Kapitel 3 beschrieben, können Zertifikate verwendet werden, um die Sicherheit (A5) zu erhöhen. Um zu überprüfen, ob eine Implementierung von einem vertrauenswürdigen Anbieter angeboten wird, kann dieses hier gespeicherte Zertifikat verwendet werden. Außerdem kann mit der Hilfe des Zertifikates festgestellt werden, ob ein Anbieter diese Implementierung ändern oder löschen darf. Wie zuvor beschrieben, können dafür Zertifikate, die beispielsweise von TÜV SÜD vergeben werden, verwendet werden. Dieser bietet eine Prüfung der Softwarequalität an, die dann mit Zertifikaten nachgewiesen werden kann. So kann beispielsweise das in [TÜV23] erklärte Zertifikat zur Zertifizierung der Cybersicherheit im Automobilbereich verwendet werden.

4.5 Aufbau des Metadatenmodells

Um das Suchen nach PETs zu ermöglichen, wurden zuvor einige benötigte Metadatenfelder identifiziert. Aus diesen wurde ein Metadatenmodell erstellt, welches diese Metadatenfelder sammelt und so eine Beschreibung der einzelnen PETs ermöglicht. Dieses Metadatenmodell wird im Folgenden vorgestellt.

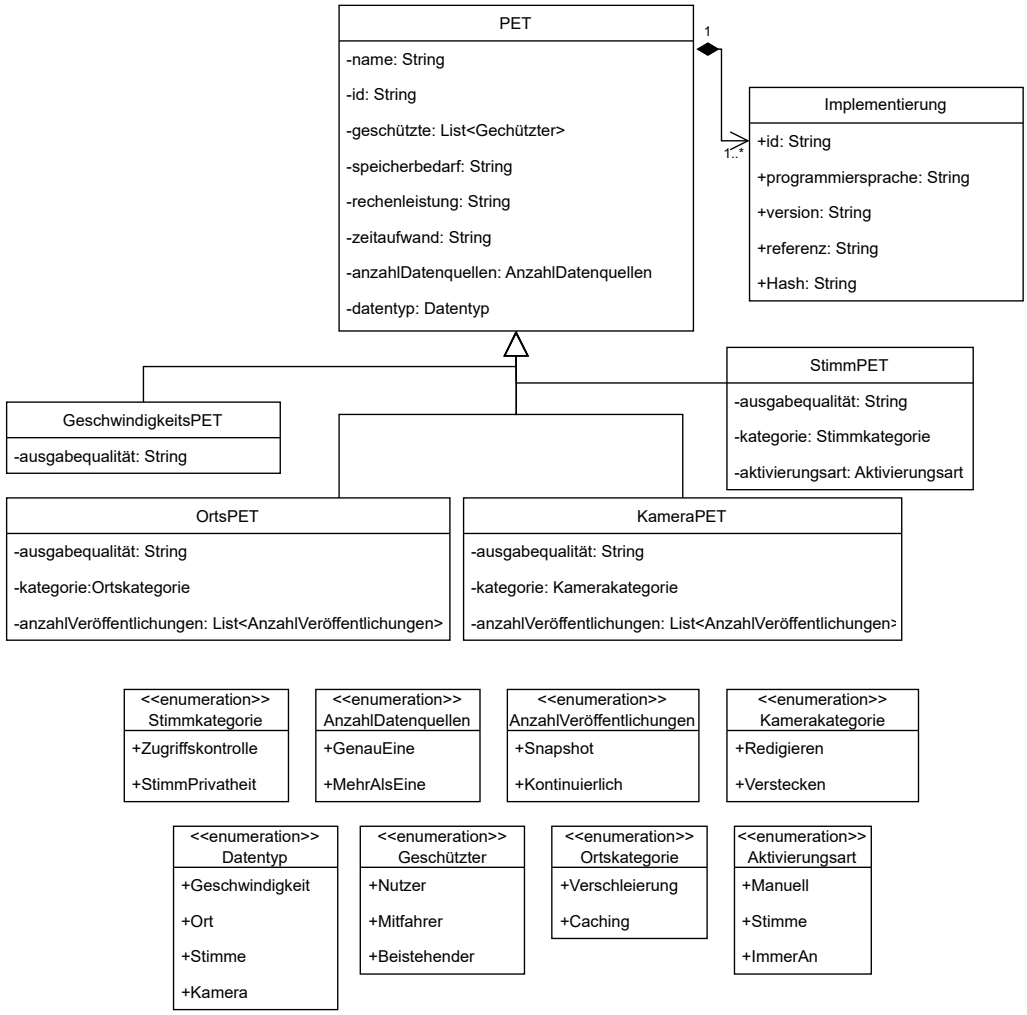


Abbildung 4.4: Metadaten Übersicht.

In Abbildung 4.4 wird das entwickelte Metadatenmodell als UML-Klassendiagramm, wie es beispielsweise von Becker et al. [BPV12] beschrieben wird, dargestellt. Dieses wurde gewählt, da es anschaulich die Beziehungen zwischen den allgemeinen und spezifischen Metadatenfeldern zeigen kann. Im Folgenden wird der Aufbau dieses Modells vorgestellt und auf die Werte eingegangen, die ein Metadatenfeld annehmen kann.

Die allgemeinen Metadaten werden in der Klasse „PET“ dargestellt. Da alle PETs diese Metadatenfelder besitzen, wird diese Klasse als Überklasse verwendet, von der die PETs der einzelnen Datentypen erben. Dadurch haben beispielsweise die PETs für Ortsdaten alle allgemeinen Meta-

datenfelder und zusätzlich noch die spezifischen Metadatenfelder. Dabei haben die PETs, alle das Metadatenfeld „Ausgabequalität“, welches zuvor noch nicht beschrieben wurde. Dieses ist nötig, da die Ausgabe eine PET bestimmt, welche Funktionen noch ausgeführt werden können. Diese Information ist wichtig für den Anfragetyp 3.

Ausgabequalität. Die Ausgabequalität beschreibt, für welche Funktionen die Ausgabe eines PETs verwendet werden kann. Dazu kann beispielsweise für Ortsdaten angegeben werden, wie nahe, die verschleierten Ortsdaten, an den ursprünglichen Daten sind. Dabei ist die Qualität von der Art der Inputdaten abhängig von den Datentypen und von der Funktion die ausgeführt werden soll und wird daher für die Datentypen einzeln spezifiziert. Für Ortsdaten kann die Ausgabequalität beispielsweise bedeuten, um wie viel eine berechnete Route länger wird, wenn statt der Originaldaten die Ausgabe des PET verwendet wird. Sie könnte aber auch beschreiben, mit welcher Wahrscheinlichkeit beispielsweise eine Tankstelle tatsächlich das näheste zum tatsächlichen Standort des Nutzers ist. Die genaue Auswahl der Kategorien wird für weitere Entwicklungen an PriTeX offen gelassen und kann in Zukunft hinzugefügt werden. Daher wird hier ein String als Platzhalter angegeben.

Die Metadatenfelder für die Implementierungen werden in der Klasse „Implementierung“ gespeichert. Dabei wird durch die verwendete Komposition zwischen der Klasse „PET“ und „Implementierung“ dargestellt, dass die Existenz einer Implementierung von der Existenz des zugehörigen PETs abhängen soll. Des Weiteren ist zu sehen, dass zu einem PET mindestens eine Implementierung gehört. Eine Implementierung gehört dabei immer zu genau einem PET.

Die Metadatenfelder, deren mögliche Werte nicht näher beschrieben wurden, können in Form eines Strings gespeichert werden. Dabei gilt für die „ID“ eines PET, dass sie eindeutig sein muss. Auch der String, der die „ID“ einer Implementierung speichert, muss eindeutig sein.

Für die Metadatenfelder, für die genaue Werte festgelegt wurden, werden Enumerationen verwendet, um die Werte, aus denen ausgewählt werden kann, festzulegen. So kann beispielsweise für die Kategorien von PETs für Ortsdaten zwischen „Verschleierung“ und „Caching“ ausgewählt werden.

Des Weiteren werden die geschützten Personen und die Anzahl der Veröffentlichungen als Liste gespeichert, da mehr als eine Art von Personengruppe durch ein PETs geschützt werden kann.

5 Implementierung des Metadatenmodells

Im vorherigen Kapitel wurde ein Konzept für das Metadatenmodell beschrieben. In diesem Kapitel soll nun eine mögliche Implementierung für das entwickelte Konzept gegeben werden. Diese soll zeigen, dass es möglich ist, das Metadatenmodell in der Praxis zu verwenden. Dazu wird zunächst beschrieben, wie das Metadatenmodell implementiert wurde. Danach wird eine Möglichkeit gegeben, wie mithilfe des Metadatenmodells nach einem PET gesucht werden kann.

5.1 Umsetzung des Metadatenmodells

Um die Metadaten überschaubar zu speichern, wird hier eine semistrukturierte Datenbank verwendet. Dafür werden ein XML-Schema und XML-Dateien verwendet [W3C16]. Dabei können mehrere XML-Dateien zu einer Datenbank zusammengefügt werden. Dazu gibt es bereits einige Programme, die das Aufbauen einer Suche aus XML-Dateien ermöglichen. Diese können für die Suche nach PETs verwendet werden. Wie das für diese Arbeit umgesetzt wird, wird in Abschnitt 5.2 erläutert.

Zunächst wird das Metadatenmodell der PETs mit einem XML-Schema beschrieben. Ein Ausschnitt dieses Schemas wird in Listing 5.1 gezeigt. Darin sieht man, wie einige der im Konzept entworfenen Metadatenfelder im XML-Schema umgesetzt werden. Ein PET wird im Schema als ein Element mit dem Complextype „tPET“ (Zeile 5-18) angegeben. Die Menge der in diesem Complextype enthaltenen Elemente stellen die Metadatenfelder dar. Die Elemente werden wiederum von Complextypes und Simpletypes definiert. Die Metadaten für eine Implementierung werden im Complextype „tImpl“ (Zeile 39-48) festgelegt. Das bedeutet, dass die verschiedenen Implementierungen nicht durch eigene XML-Datei beschrieben, sondern werden in der gleichen XML-Datei gespeichert, in der auch die Metadaten des zugehörigen PET gespeichert werden. Wird eine neue Implementierung hinzugefügt oder eine alte gelöscht, so muss die XML-Datei des PET aktualisiert werden. Die für einen Datentyp spezifischen Metadatenfelder werden im Compeltype „tDataType“ (Zeile 27-37) festgelegt. Dabei kann zunächst zwischen den vier Datentypen ausgewählt werden. Innerhalb dieses Elements, können dann die weiteren Metadaten, die für diesen Datentyp spezifisch sind, eingetragen werden. So wird die in Abbildung 4.4 dargestellte Vererbung der Metdatenfelder umgesetzt. Die im UML-Diagramm (Abbildung 4.4) verwendeten Enumerationen werden im XML-Schema durch Simpletypes beschrieben. Ein Beispiel dafür ist der in Zeile 20-25 stehende Simpletype „tNumberOfSources“.

Wie beschrieben sollen die Identifikatoren der PETs und ihrer Implementierungen eindeutig sein. Dazu wird im XML-Schema dadurch getan, dass die „ID“ jeweils als „xs:ID“ (Zeile 9 und 42) festgelegt wird. Dadurch wird sichergestellt, dass jeder Wert eines solchen Identifikators nur genau einmal in einem Dokument vorkommt. Da diese Eindeutigkeit nur innerhalb eines Dokumentes gilt, kann dadurch hier die Eindeutigkeit nicht nur damit umgesetzt werden. Das liegt daran, dass hier nicht eine große XML-Datei, sondern wie beschrieben mehrere kleinere XML-Dateien

Listing 5.1 Ausschnitt aus den XML-Schema für die Metadaten

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="PET" type="tPET"/>

  <xs:complexType name="tPET">
    <!-- Allgemeine Metadaten eines PETs -->
    <xs:sequence>
      <xs:element name="Name" type="xs:string" />
      <xs:element name="ID" type="xs:ID"/>
      <xs:element name="ProtectedPeople" type="tProtectedPeople"/>
      <xs:element name="MemoryRequirement" type="xs:string"/>
      <xs:element name="TimeRequirement" type="xs:string"/>
      <xs:element name="ProcessingPower" type="xs:string"/>
      <xs:element name="NumberOfSources" type="tNumberOfSources"/>
      <xs:element name="DataType" type="tDataType"/>
      <xs:element name="Implementation" type="tImpl" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:simpleType name="tNumberOfSources">
    <xs:restriction base="xs:string">
      <xs:enumeration value="ExactlyOne"/>
      <xs:enumeration value="MoreThanOne"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="tDataType">
    <!-- Auswahlmöglichkeiten für den Datentyp -->
    <xs:sequence>
      <xs:choice>
        <xs:element name="Speed" type="tSpeed"/>
        <xs:element name="Location" type="tLocation"/>
        <xs:element name="Voice" type="tVoice"/>
        <xs:element name="Camera" type="tCamera"/>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
  ...
  <xs:complexType name="tImpl">
    <!-- Metadaten der Implementierung -->
    <xs:sequence>
      <xs:element name="ID" type="xs:ID"/>
      <xs:element name="ProgrommingLanguage" type="xs:string"/>
      <xs:element name="Version" type="xs:string"/>
      <xs:element name="Reference" type="xs:string"/>
      <xs:element name="Hash" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  ...
</xs:schema>
```

Listing 5.2 Beispiel einer XML-Datei. Einträge orientiert an dem in [ALH+11] vorgestellten Caché. Die nicht direkt besprochenen Eigenschaften von Caché wurden mit Beispielwerten gefüllt.

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <PET xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="
   PETSchema2.xsd">
3    <Name>Cache</Name>
4    <ID>fuiebff</ID>
5    <ProtectedPeople>
6      <ProtectedPerson>User</ProtectedPerson>
7      <ProtectedPerson>Passenger</ProtectedPerson>
8    </ProtectedPeople>
9    <MemoryRequirement>560</MemoryRequirement>
10   <TimeRequirement>Zeitaufwand</TimeRequirement>
11   <ProcessingPower>Rechenzeit</ProcessingPower>
12   <NumberOfSources>ExactlyOne</NumberOfSources>
13   <DataType>
14     <Location>
15       <OutputQuality>Qualität</OutputQuality>
16       <Category>Caching</Category>
17       <NumberOfPublicationsList>
18         <NumberOfPublications>Snapshot</NumberOfPublications>
19         <NumberOfPublications>Continuous</NumberOfPublications>
20       </NumberOfPublicationsList>
21     </Location>
22   </DataType>
23   <Implementation>
24     <ID>hfhfs</ID>
25     <ProgrommingLanguage>Java</ProgrommingLanguage>
26     <Version>1</Version>
27     <Reference>Pfad/zu/Chache</Reference>
28     <Hash>fge</Hash>
29   </Implementation>
30 </PET>
```

verwendet werden, um die Metadaten der PETs zu speichern. Daher muss ein weiterer Mechanismus verwendet werden, um die Eindeutigkeit der Identifikatoren zu gewährleisten. Das Backend kann dafür eingesetzt werden. Dieses kann vor dem Einfügen eines neuen PETs überprüfen, ob die angegebenen Identifikatoren bereits verwendet wurden und so denjenigen, der das PET hinzufügen möchte, darüber informieren, dass er eine andere ID wählen muss, um das PET einfügen zu können.

Aus dem XML-Schema können dann die XML-Dateien für die einzelnen PETs abgeleitet werden. Diese XML-Dateien können dann gemeinsam mit den Implementierungen der PETs im Repository gespeichert werden. Ein Beispiel einer solchen XML-Datei für ein PET wird in Listing 5.2 dargestellt. Zu sehen ist dort, wie aus dem XML-Schema eine XML-Datei entsteht. Dabei wurden die in Tabelle 4.1 vorgestellten Werte von Caché [ALH+11] verwendet. Dort sieht man, wie die Auswahl des Datentypen die weiteren Metadatenfelder beeinflusst. Auch zu sehen ist, dass für das PET eine Implementierung beschrieben wird. Da in der Beschreibung von Caché diese nicht genau beschrieben wird, wird hier eine Implementierung mit Beispielwerten gefüllt.

5.2 Aufbau der Suchtabelle

Nachdem zuvor festgelegt wurde, wie die Daten gespeichert werden, soll nun die Suche ermöglicht werden. Um das Umzusetzen wird XQuery 3.1 [W3C17] verwendet. Mit dieser Abfragesprache können XML-Datenbanken beziehungsweise XML-Dateien durchsucht werden.

Da die Metadaten der PET wie beschrieben verteilt gespeichert werden sollen und können, ist es nötig einen Startpunkt festzulegen, von welchem man alle XML-Dateien finden kann. Um die Adressen zu einzelnen PETs XML-Dateien zu speichern, wird eine weitere XML-Datei verwendet, in welcher die PETs als Elemente vorkommen. In diesen Elementen ist jeweils die Referenz zu der XML-Dateien, die die Metadaten der PET enthält, als Attribut enthalten. Diese XML-Datei wird im folgenden als Katalog bezeichnet. Ein Beispiel für einen solchen Katalog wird durch Listing 5.3 gegeben.

Listing 5.3 XML-Datei des Katalogs von PETs mit Beispiel PETs

```
<?xml version="1.0" encoding="UTF-8"?>
<Pets>
  <PET reference="SpAn.xml"/>
  <PET reference="LoKa.xml" />
  <PET reference="TQ.xml" />
  <PET reference="CaDaA.xml" />
</Pets>
```

Mit diesem Katalog ist nun der Startpunkt für die Suche gegeben. Eine solche Anfrage kann wie in Listing 5.4 aussehen. Dazu wird auf die dort gespeicherten Referenzen zugegriffen, wodurch die Pfade zu den weiteren XML-Dateien herausgefunden werden. Mit diesen kann dann weiter gefiltert werden.

Listing 5.4 XQuery Abfrage, welche im Katalog vorhandene PETs zurückgibt

```
for $x in doc("Katalog.xml")/Pets/PET/@attribute(reference)
return $x
```

So können die in Abschnitt 4.1 festgelegten Anfragetypen umgesetzt werden. Möchte man beispielsweise alle PETs finden, die einen Schutz für die Ortsdaten geben, dann kann das mit folgender Abfrage, in Listing 5.5, geschehen. Diese Anfrage entspricht dem in Abschnitt 4.1 definierten Anfragetyp 1. In der Abfrage werden alle PETs zurückgegeben, in denen das Feld „Location“ existiert. Da dieses Feld nur bei PETs vorhanden ist, die den Datentyp Ortsdaten verschleiern, können so die gewünschten PETs herausgefiltert werden.

Listing 5.5 XQuery Abfrage, welche im Katalog vorhandene PETs, die Ortsdaten schützen, zurückgibt

```
for $x in doc("P:\Pfad\zu\Ordner\Katalog.xml")/Pets/PET/@attribute(reference)
where (fn:exists(doc(fn:concat("P:\Pfad\zu\Ordner\", xs:string($x)))/PET/DataType/Location))
return $x
```

Möchte man stattdessen eine Anfrage vom Typ 2, aus Abschnitt 4.1, tätigen und ein PET für die Ausführung auf dem eigenen Fahrzeug finden, so muss man zunächst die Ressourcen, die im Fahrzeug sind kennen. Außerdem darf dann wie zuvor beschrieben die Anzahl der Datenquellen nur genau eine sein. Eine solche Anfrage wird in Listing 5.6 dargestellt. Dort wird sowohl nach der Anzahl der Datenquellen und dem Speicheraufwand gefiltert. Es wird davon ausgegangen, dass der Speicherbedarf in der Form einer ganzen Zahl gespeichert wurde und das Fahrzeug einen Speicherplatz bis 570 hat.

Listing 5.6 XQuery Abfrage, welche im Katalog vorhandene PETs, die Ortsdaten schützen, zurückgibt

```
for $x in doc("P:\Pfad\zu\Ordner\Katalog.xml")/Pets/PET/@attribute(reference)
where (
  (doc(fn:concat("P:\Pfad\zu\Ordner\", xs:string($x)))/PET/NumberOfSources/text()
    eq
  "ExactlyOne")
  and
  (xs:integer(doc(fn:concat("P:\Pfad\zu\Ordner\",xs:string($x)))/PET/MemoryRequirement/text())
    <
  xs:integer(570)))
return $x
```

Die Anfrage von Typ 3 ermöglicht, wie in Abschnitt 4.1 beschrieben, die Suche nach einem PET, welches Schutz bietet, aber auch die Nutzung einer bestimmten Funktion zulässt. Möchte man also beispielsweise einen Routenplaner nutzen, der kontinuierlich den Standort abfragt, und so den Nutzer zu seinem Ziel leitet, so muss ein PET gefunden werden, dass kontinuierlich Daten verarbeiten kann und die Funktion weiterhin zulässt. Da die Ausgabequalität in dieser Arbeit nicht genau definiert wurde, beschränkt sich die Beispielanfrage in Listing 5.7 auf die Anzahl der Veröffentlichungen.

Listing 5.7 XQuery Abfrage, welche im Katalog vorhandene PETs, die Ortsdaten schützen, zurückgibt

```
let $publicationEquals := function($x as xs:string, $publication as xs:string){
  for $y in doc(fn:concat("P:\Pfad\zu\Ordner\", xs:string($x)))/PET/DataType/Location/
NumberOfPublicationsList/NumberOfPublications
  where $y eq $publication
  return true()
}

for $x in doc("D:\Uni\Bachelorarbeit\TexProjekt\Vorlage_BA_Egeler\BA\graphics\xquery\Catalog.
xml")/Pets/PET/@attribute(reference)
where (
  (fn:exists(doc(fn:concat("D:\Uni\Bachelorarbeit\TexProjekt\Vorlage_BA_Egeler\BA\graphics\
xquery\", xs:string($x)))/PET/DataType/Location ))
  and
  ($publicationEquals($x, "Continuous")))
return $x
```

5.3 Suche mithilfe der graphischen Oberfläche

Da die graphische Oberfläche, an der Interaktion des Nutzers mit PriTeX, wesentlich beteiligt ist, wird hier eine beispielhafte Suche mithilfe der graphischen Oberfläche gegeben. In Kapitel 3 wurde bereits beschrieben, welche der in Abschnitt 2.3 Anforderungen mithilfe der graphischen Oberfläche umgesetzt werden sollen. Diese Anforderungen sind die Auffindbarkeit (A1) und die Verständlichkeit (A3). Die graphische Oberfläche soll es also ermöglichen, sich einen Überblick über die gespeicherten PETs zu machen und die Arbeitsweise der PETs zu verstehen. Für eine Demonstration der möglichen Umsetzung dieser Anforderungen mit der graphischen Oberfläche wurde ein Design für diese entworfen.

Für die beispielhafte Suche wird davon ausgegangen, dass der Nutzer eine Anfrage des in Abschnitt 4.1 beschriebenen Anfragetyp 3 durchführen will. Der hier angenommene Nutzer möchte einen Dienst verwenden, der nach dem Übermitteln des aktuellen Standorts Restaurants in der Nähe vorschlägt. Dieser Dienst wird von einem Drittanbieter angeboten. Da er dem Drittanbieter seinen genauen Standort nicht weiterleiten möchte, ist er auf der Suche nach eine PET, mit welchem er seinen Standort schützen kann.

Zu Beginn der Suche zeigt die graphische Oberfläche zunächst mögliche Filter für die Suche an. Diese Ansicht wird in Abbildung 5.1 dargestellt. Zunächst muss ausgewählt werden, welcher Datentyp geschützt werden soll. Dadurch können die Filter zu den spezifischen Metadatenfeldern zu diesem Datentyp angezeigt werden.Der Nutzer wählt also zunächst „Ortsdaten“ für den Datentyp aus, da er wie zuvor beschrieben seinen Standort schützen möchte. Dafür klickt er mit dem Mauszeiger auf das Feld „Ortsdaten“.

Abbildung 5.1: Mögliches Design der graphischen Oberfläche - Auswahl des Datentyps.

Nun bildet die graphische Oberfläche die möglichen Filter für diesen Datentyp, sowie eine Liste der möglichen PETs, ab. Die möglichen Filter sind in Abbildung 5.2 zu sehen. Da der Nutzer sich selbst schützen möchte, wählt er „Nutzer“ für die zu schützenden Personen aus. Auch die

Anzahl der Datenquellen wird mit dem gleichen Grund ausgewählt. Das bedeutet, dass er für die der Datentypen „GenauEine“ auswählt. Da er wie beschrieben eine Funktion nutzen möchte, die als Eingabe einmalig den Standort des Nutzers erwartet, wählt er für die Anzahl der Veröffentlichungen „Snapshot“ aus. Dann klickt er auf „Filter anwenden“.

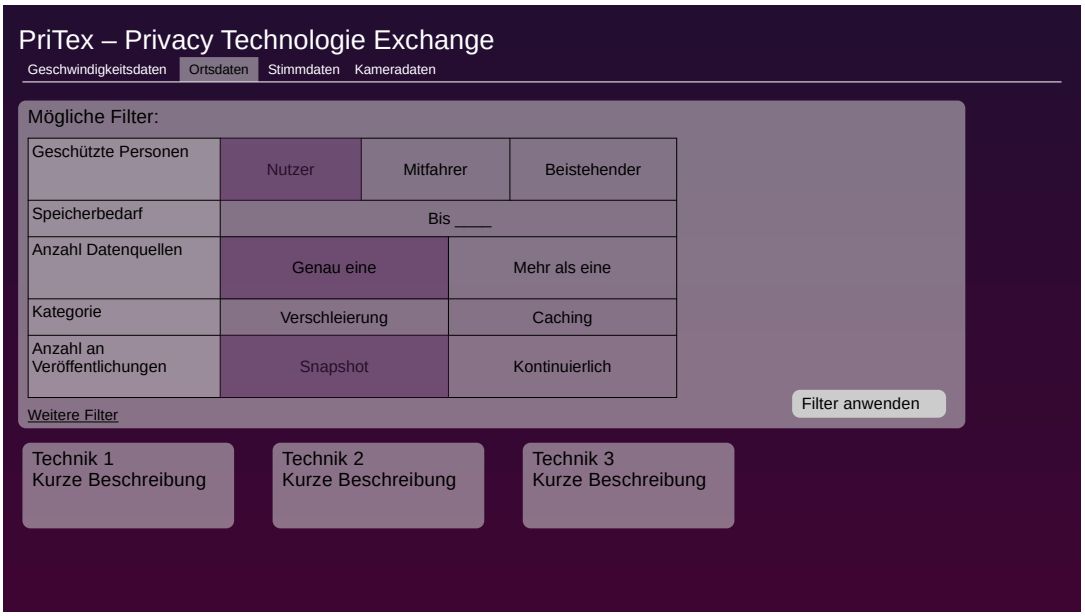


Abbildung 5.2: Mögliches Design der graphischen Oberfläche - Filterung.

Jetzt kann der Nutzer aus den abgebildeten PETs eines auswählen. Dazu klickt er auf eines der unter den Filtern angezeigten PETs. Dadurch wird eine neue Ansicht geöffnet, in dem die Eigenheiten der Technik beschrieben werden. In Abbildung 5.3 wird eine Beispielansicht einer solchen Informationsseite über ein PET gezeigt. Dort wird eine Beschreibung der Arbeitsweise eines PETs gegeben. Diese Beschreibung orientiert sich an den in Metadaten gespeicherten Eigenschaften eines PETs. Durch einen Experten im Bereich der Privatsphäre kann ein solcher Text erstellt werden. Auch können hier Illustrationen von den Originaldaten und den verschleierten Daten gegeben werden. Dadurch können die PETs und ihre Arbeitsweise verständlich (A3) gemacht werden.

Durch diese beispielhafte Suche wurde gezeigt, wie die graphische mithilfe des Metadatenmodells und des Backends die Suche nach bestimmten PETs umsetzt. Dabei wurde beispielhaft eine Anfrage des Anfragetyp 3 durchgeführt. Die anderen Anfragetypen können in analog durchgeführt werden. Dadurch wurde gezeigt, dass die Anforderung der Auffindbarkeit (A1) durch PriTeX umgesetzt werden kann.

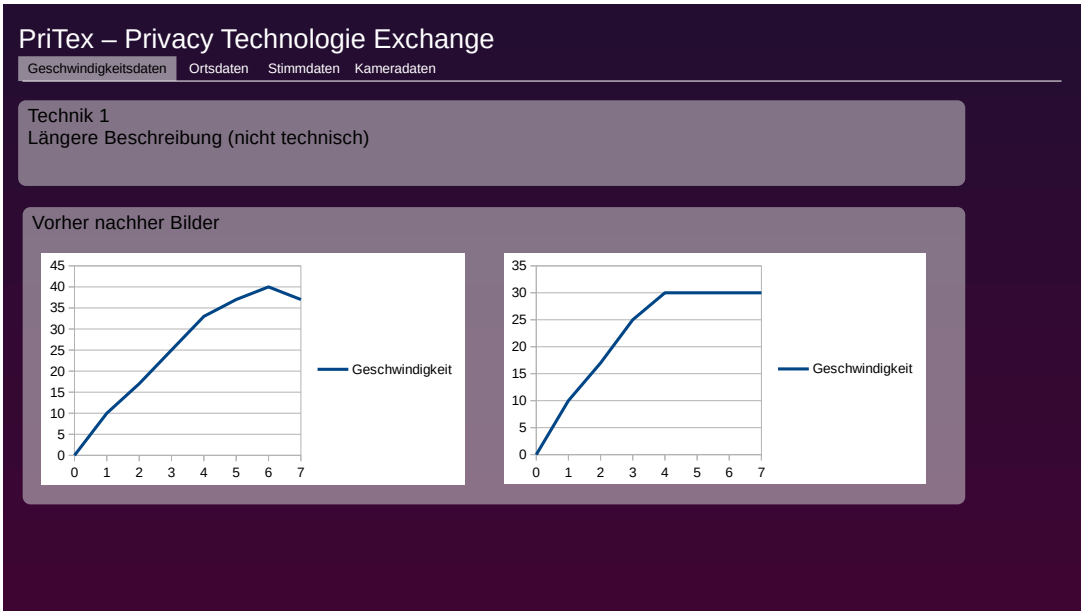


Abbildung 5.3: Mögliches Design der graphischen Oberfläche - Informationsseite für ein PET.

6 Evaluation und Diskussion der Ergebnisse

Nachdem im vorherigen ein Konzept und ein die Implementierung des Metadatenmodells der vorgeschlagenen Lösung gegeben wurde, ist es nun wichtig zu evaluieren, ob die Lösung das Problem tatsächlich lösen kann. Dazu soll in diesem Kapitel gezeigt werden, wie das entworfene System PriTeX die Anforderungen erfüllt. Zunächst wird beschrieben, wie die Anforderungen aus dem Datamesh (Abschnitt 2.3) umgesetzt wurden. Dabei wird zuerst auf das Konzept von PriTeX eingegangen, gefolgt von der Evaluation der Anforderungen an das Metadatenmodell. Die Anforderungen, zu denen bereits zu Beginn festgestellt wurde, dass sie hier nicht umsetzbar sind oder bereits durch die Definition der PETs umgesetzt sind, werden dabei nicht berücksichtigt. Danach werden die Limitierungen des entwickelten Systems beschrieben.

6.1 Evaluierung von PriTeX

Zunächst soll das Konzept von PriTeX evaluiert werden. Dafür wird die Umsetzung der in Abschnitt 2.3 vorgestellten und in Kapitel 3 den Komponenten von PriTeX zugeordneten Anforderungen evaluiert.

Anforderung 1: Auffindbarkeit. Die Auffindbarkeit der PETs, bezieht sich darauf, dass es möglich sein soll, sich einen Überblick über die gespeicherten PETs zu machen. Sie wird durch die graphische Oberfläche und das Metadatenmodell ermöglicht. Die graphische Oberfläche kann dafür eine Übersicht geben, über die Kategorien von PETs, die im System gespeichert sind beziehungsweise gespeichert werden können. Dadurch ist es dem Nutzer möglich, die im System vorhandenen PETs zu überblicken. Ein Beispiel einer solchen Ansicht wurde in Abbildung 5.3 gegeben. Diese Ansicht kann für die verschiedenen PET erstellt werden und die wichtigen Informationen über das PET darstellen. Die Verwendung des Metadatenmodells und die daraus entstehende Verbesserung der Auffindbarkeit werden in Abschnitt 6.2 erläutert.

Anforderung 3: Verständlichkeit. Die Verständlichkeit der PETs beschreibt, dass es einem Nutzer möglich sein soll, die Arbeitsweise und den Schutz eines PETs verstehen zu können. Das Konzept unterstützt die Verständlichkeit der im System enthaltenen PETs. Dazu kann mit der graphischen Oberfläche eine Beschreibung der PETs gegeben werden, welche die Arbeitsweise, die erwartete Eingabe, die Art der ausgegebenen Daten und weitere wichtige Informationen zu einem PET an den Nutzer weiter gibt. Diese Beschreibung der PETs wurde in Abschnitt 5.3 genauer beschrieben. Durch soll der Nutzer verstehen, welche Vor- und Nachteile der Einsatz eines bestimmten PET mit sich bringt und auch die Arbeitsweise der PETs nachvollziehen. Dazu kann auch das Metadatenmodell mit verwendet werden. Die Evaluierung der Umsetzung durch das Metadatenmodell wird in Abschnitt 6.2 beschrieben.

Anforderung 5: Sicherheit. Die Sicherheit bezieht sich auf den Zugriff auf PriTeX und die dort gespeicherten PETs. Das Konzept ermöglicht dem System das Umsetzen dieser Anforderung. Dazu wird das Backend eingesetzt. Dieses verwendet Zertifikate verwenden, mit welchen der Zugriff auf das Repository gesteuert wird. Dadurch kann sichergestellt werden, dass nur vertrauenswürdige Personen oder Firmen PETs hinzufügen, verändern oder löschen dürfen. Für die Speicherung der Zertifikate wird das Metadatenmodell verwendet. Die Hilfe des Metadatenmodells bei der Umsetzung der Sicherheit wird in Abschnitt 6.2 beschrieben.

Anforderung 6: Native Zugänglichkeit. Die native Zugänglichkeit beschreibt, dass die gespeicherten PETs in verschiedenen Fahrzeugen durchgeführt werden können. Sie wird durch das Konzept von PriTeX umgesetzt. Um sie zu erreichen, können zu einem PETs mehrere unterschiedliche Implementierungen im Repository gespeichert werden. Dadurch können verschiedene Fahrzeuge mit unterschiedlichen Betriebssystemen das PET verwenden. Die Information, welche Implementierungen in welchem Fahrzeug ausgeführt werden können, wird nicht explizit gespeichert, kann jedoch aus den gespeicherten Metadaten, wie dem Zeitaufwand oder der verwendeten Programmiersprache, herausgelesen werden.

6.2 Evaluierung des Metadatenmodells

In diesem Abschnitt wird die Umsetzung der Anforderungen an das Metadatenmodell beschrieben. Zu Beginn dieser Arbeit wurden in Abschnitt 2.3 bereits die Anforderungen, die sich vom Data Mesh ableiten lassen, beschrieben. Das diese umgesetzt wurden, wird im Folgenden begründet. Dabei wird auf die in der Implementierung umgesetzten Lösungen eingegangen.

Anforderung 1: Auffindbarkeit. Das Metadatenmodell gibt ein Schema, mit dem die PETs beschrieben werden können. Dadurch kann eine Suche, basierend auf diesen Informationen realisierte werden. Dazu wurden in Abschnitt 4.1 Anfragetypen definiert, die von einem Nutzer gemacht werden können. Auch kann durch das Metadatenmodell nach den PETs gesucht werden, wodurch diese auch leichter auffindbar sind. Die Suche mit der Hilfe dieser Anfragetypen wurde in Abschnitt 5.2 gezeigt. Es ist also möglich, mit der Hilfe des entwickelten Metadatenmodells. Wie zuvor beschrieben ist auch die graphische Oberfläche an der Umsetzung der Auffindbarkeit beteiligt. Zusammen ermöglichen das Metadatenmodell und die graphische Oberfläche die Suche nach PETs und setzen damit die Auffindbarkeit der PETs um.

Anforderung 2: Adressierbarkeit. Für die Adressierbarkeit sollen die PETs eindeutig identifiziert und adressiert werden. Dabei wird sie durch das Metadatenmodell umgesetzt. Die gespeicherten Metadaten zu einem PET dienen dabei die Adresse dieses PETs. Um diese Adresse eindeutig zu machen, wird wie beschrieben ein Metadatenfeld für den Identifikator eingesetzt. Dieser Identifikator ist eindeutig und keine zwei PETs haben den gleichen Identifikator. Auch wenn neue Versionen eines PET oder neue Implementierungen eingefügt werden, bleibt diese Adresse eindeutig. Das liegt daran, dass die grundsätzlichen Eigenschaften des PETs durch solche Änderungen gleich bleiben. Des Weiteren werden die Metadaten der Implementierungen eines PETs zusätzlich zu der Beschreibung des zugehörigen PETs gespeichert. Dadurch können auch die zu den PETs gespeicherten Implementierungen können eindeutig adressiert werden. Sie besitzen dafür ebenfalls einen eindeutigen Identifikator. Insgesamt ermöglicht das Metadatenmodell also eine Adressierung für die PETs und ihre Implementierungen.

Anforderung 3: Verständlichkeit. Zuvor wurde in Abschnitt 6.1 bereits evaluiert, dass die graphische Oberfläche für die Umsetzung der Verständlichkeit verwendet werden kann. Auch das Metadatenmodell unterstützen die Verständlichkeit der im System enthaltenen PETs. Dazu wird mit der graphischen Oberfläche eine Beschreibung der PETs gegeben, welche die Arbeitsweise, die erwartete Eingabe, die Art der ausgegebenen Daten und weitere wichtige Informationen zu einem PET an den Nutzer weiter gibt. Durch diese kann der Nutzer verstehen, welche Vor- und Nachteile der Einsatz eines bestimmten PET mit sich bringt und auch die Arbeitsweise der PET nachvollziehen, wodurch die PETs verständlich werden.

Anforderung 4: Vertrauenswürdigkeit und Ehrlichkeit. Die Vertrauenswürdigkeit der Implementierung eines PET beschreibt, ob ein PET aus einer vertrauenswürdigen Quelle kommt und nicht verändert wurde. Dieses Vertrauen wird mithilfe des Metadatenmodells etabliert werden. Dort werden die nötigen Daten, also ein Hash, gespeichert werden. Diesen verwendet der Nutzer, um zu überprüfen, ob eine Implementierung der in PriTeX gespeicherten Implementierung entspricht. Dadurch kann der Nutzer sicher gehen, dass er die Implementierung von einer vertrauenswürdigen Quelle bekommen hat.

Anforderung 5: Sicherheit. Diese Anforderung beschäftigt sich mit der Verhinderung des unautorisierten Zugriffs auf PriTeX und die darin gespeicherten PETs. Zuvor wurde in Abschnitt 6.1 beschrieben, wie das Backend die Umsetzung dieser Anforderung ermöglicht. Wie beschrieben werden dafür Zertifikate verwendet. Das Metadatenmodell ermöglicht dabei den Zugriff auf PETs zu steuern. Es sieht vor, dass zu jedem PET ein Zertifikat gespeichert wird. Wird versucht, mit einem anderen Zertifikat auf das PET zuzugreifen, so wird der Zugriff abgelehnt. Dadurch wird der Zugriff auf das PET gesteuert und damit die Anforderung der Sicherheit umgesetzt.

6.3 Einschränkungen

In diesem Abschnitt werden die Einschränkungen des entwickelten Systems beschrieben. Dabei wird auf die Problematiken eingegangen, die sich durch das Konzept und die Implementierung ergeben und auch auf solche, die das Szenario mit sich bringt.

In PriTeX können, wie zuvor erwähnt, nur PETs gespeichert werden, die die Privatsphäre mit der Hilfe von Software schützen. Dadurch ist es nicht möglich, seine Privatsphäre zu schützen, wenn man schon dem Endgerät nicht vertraut. Da in dieser Arbeit davon ausgegangen wird, dass das eigene CV vertrauenswürdig ist, wird diese Problematik hier nicht betrachtet. Dennoch gibt es Szenarien, in denen das wichtig ist und welche Hardwarelösungen verlangen. Möchte man sich selbst beispielsweise davor schützen, dass einen die Mikrofone von anderen Geräten aufzeichnen können, so kann man eine Hardware verwenden, die die Aufnahme dieser Mikrofone stört. Eine solche Lösung wurde beispielsweise von Chen et al. [CLT+20] gegeben. Sie entwickelten ein tragbares Gerät, mit dem sich die Mikrofone in der Umgebung gestört, sodass diese keine nutzbaren Aufnahmen machen kann. Das Einfügen von Hardwarelösungen in PriTeX könnte durch folgende Lösung ermöglicht werden. PriTeX speichert die Beschreibung dieser PETs mit Hardwarelösungen und beschreibt zusätzlich, welche Hardware dafür verwendet wird und kann zum Beispiel auf eine Webseite verweisen, über die man diese Hardware bestellen kann. So könnten Anbieter solcher PETs eine Beschreibung dieser PETs speichern und Nutzer die PETs finden.

Der bisher implementierte Teil der Lösung hat nur eingeschränkte Funktionalität. Er ermöglicht die Speicherung der Metadaten und die Suche nach den gespeicherten PETs. Andere Anforderungen wurden bisher nicht umgesetzt. So zum Beispiel die Vertrauenswürdigkeit (A4) einer Implementierung. Da bisher nur die Suche nach PETs implementiert wurde und den nicht die Implementierungen gespeichert und beschrieben wurden, wurde auch noch nicht der Hash für diese Implementierungen festgelegt. Dadurch kann der Nutzer noch keine Implementierungen aus PriTeX herunterladen oder überprüfen, ob eine Implementierung vertrauenswürdig ist. Durch die Implementierung der im Konzept vorgeschlagenen Komponenten können diese Anforderung jedoch umgesetzt werden.

Durch PriTeX wird den Nutzern die Möglichkeit gegeben, ihre Privatsphäre, bei der Nutzung eines CVs, zu schützen. Dadurch ist jedoch nicht sichergestellt, dass ein Nutzer sich für den Einsatz eines PETs oder eines anderen Schutzes, entscheidet. Denn auch wenn viele Personen ihre Privatsphäre schützen wollen, werden doch sehr viele sensible Daten geteilt. Dieses Phänomen wird als Privatsphäre Paradoxon bezeichnet und wurde beispielsweise von [NHH07] beschrieben. Dabei ist die Problematik, dass eine Lücke zwischen den Intentionen und den tatsächlich geteilten Daten vorhanden ist. Personen teilen also Daten, obwohl sie eigentlich Bedenken haben. Diese Problematik kann PriTeX nicht vollständig lösen, sondern es kann den Leuten, die bereits auf der Suche nach einem Schutz sind, das Finden eines passenden Schutzes erleichtern. PriTeX könnte, mit der Hilfe der graphischen Oberfläche den Nutzern mitteilen, welche Gefahren, ohne den Schutz der Privatsphäre entstehen können. Dadurch kann das Problem jedoch nur teilweise gelöst werden, da ein Nutzer von PriTeX zumindest schon ein grundlegendes Interesse hat, seine Daten zu schützen oder sich zu informieren, wie er seine Daten schützen kann.

Da ein CV mehr als nur eine Person aufzeichnet, gibt es auch mehr als einen Besitzer der Daten. Selbst im hier angenommenen Szenario gibt es bereits einige verschiedene Personengruppen, die Besitzer von Daten sein können. Das sind beispielsweise der Fahrer, die Mitfahrer, Fußgänger, Fahrradfahrer, Fahrer un Mitfahrer anderer Fahrzeuge und weitere. All diese Personen haben einen Anspruch darauf, die über sie aufgezeichneten Daten einzusehen. Die DSGVO [Eur16] bestimmt zum Beispiel, dass eine Person, über die Daten gespeichert wurden, ein Recht darauf hat, diese Daten einzusehen und auch die gespeicherten Daten löschen zu lassen. Das ist bisher in Fahrzeugen nicht möglich. Um den Nutzern die Kontrolle über ihre Daten zu geben, könnte ein System entwickelt werden, mit dem einem Fahrzeug die Präferenzen der beteiligten Personen mitgeteilt werden, sodass dieses auch diese Präferenzen berücksichtigen kann. Dazu könnten die Präferenzen beispielsweise auf dem Smartphone gespeichert werden und dann per Funk an nahegelegene Fahrzeuge übertragen werden. Die Implementierung eines solchen Systems könnte in einer Erweiterung von PriTeX gespeichert werden.

7 Verwandte Arbeiten

In diesem Abschnitt werden verwandte Arbeiten genannt und beschrieben. Dabei soll das in dieser Arbeit entwickelte PriTeX gegenüber anderen Arbeiten differenziert werden. Zunächst wird auf anderes Repository für PETs eingegangen. Dieses kann mit dieser Arbeit verglichen werden, da, wie in PriTeX, PETs gespeichert werden. Des Weiteren werden andere Repositories im Bereich der Privatsphäre vorgestellt. Diese Repositories beschäftigen sich wie PriTeX mit dem Schutz der Privatsphäre der Nutzer, weswegen sie hier beschrieben werden. Danach wird auf Techniken eingegangen, mit denen sich Anforderungen an die Privatsphäre beschrieben lassen. Wie auch das Metadatenmodell können diese Beschreibungen verwendet werden, um den Schutz der Privatsphäre der Nutzer zu verbessern. Zuletzt wird eine Arbeit genannt, die ein ähnliches Problem wie PriTeX behandelt. Da sie auch auf die Problematik der zahlreichen unterschiedlichen Lösungen für den Schutz der Privatsphäre eingeht, kann sie als eine andere Art der Lösung mit PriTeX verglichen werden.

Fan und Gunja [FG20] entwickelten ein Repository für privatsphäreverbessernde Methoden für Ortsdaten. Dabei konzentrieren sie sich auf PETs, die die Daten von einer Quelle, welche eine Android-App ist, verwenden. Das Repository ist Open Source und möchte so privatsphäreverbessernde Methoden für kommerzielle und Forschungszwecke etablieren. Dabei konzentrieren sie sich auf Methoden, die die Daten eines Nutzers, also von einer Datenquelle, verschleiern. Außerdem sollen die enthaltenen Methoden einen geringen Rechen- und Speicheraufwand haben. Dadurch können die Methoden auf kleineren Geräte wie Smartphones ausgeführt werden und die Berechnung kann ohne großen Zeitverlust in den Berechnungsfluss von Funktionen eingebaut werden. Die Implementierungen der Methoden soll in einer einheitlichen Art geschehen und die bereits enthaltenen Implementierungen sind nur von der Android-App abhängig. Dadurch sind sie auf unterschiedlichen Geräten einsetzbar. Im Gegensatz zu PriTeX kann mit dem Repository ein PET nur gespeichert, aber nicht gesucht werden. Zudem ist PriTeX nicht auf PETs für Ortsdaten beschränkt, sondern betrachtet auch PETs für andere Datentypen. Auch konzentrieren sie sich nur auf PETs, die genau eine Datenquelle erwarten, während PriTeX auch PETs speichert, die mehr als eine Datenquelle erwarten. Des Weiteren beschäftigt sich PriTeX mit einem anderen Szenario. Statt PETs für Apps für Smartphone zu speichern, speichert PriTeX PETs für CVs.

Andere Repositories, die sich mit dem Schutz der Privatsphäre beschäftigen, werden im Folgenden beschrieben. Diese Repositories haben das Ziel Daten sicher zu speichern und zu teilen. Sie bieten also einen Dienst an, der das umsetzen kann, statt Techniken anzubieten, mit denen sensible Daten sicher geteilt werden können, wie es bei PriTeX der Fall ist. Die Gemeinsamkeit dieser Repositories und PriTeX ist also das Ziel Daten sicher zu teilen. Im Folgenden wird ein solches Repository genannt. Danach wird eine Lösung für das privatsphäreerhaltende Abrufen von Daten aus einem Repository beschrieben.

Das von Ding und Sato [DS20] vorgestellte Repository, welches „Derepo“ genannt wird, beschäftigt sich mit dem Szenario der Smart Health. Das Repository ist ein dezentraler Datenspeicher. Es soll sowohl die Privatsphäre als auch die Sicherheit gewährleisten. Dabei gibt es keinen zentralen Mechanismus, der den Zugriff auf die Daten bestimmt. Stattdessen wird eine Methode für die dezentralisierte Zugriffsverwaltung eingesetzt. Um die Privatsphäre der Nutzer zu schützen, wird eine homomorphe Verschlüsselung für die Daten verwendet. Auf diesen verschlüsselten Daten können dann die Berechnungen ausgeführt werden, ohne private Informationen einsehen zu können. Im Gegensatz zu PriTeX wird hier von einem anderen Szenario ausgegangen. Des Weiteren wird die Privatsphäre vor allem durch die homomorphe Verschlüsselung geschützt. Um das auch für CVs umzusetzen, müsste jedes Fahrzeug alle Daten so verschlüsseln und die Funktionen müssen auf diesen Daten die Berechnungen durchführen können. Es muss also noch eine Entwicklung dieser Funktionen geschehen. Die in PriTeX enthaltenen Methoden können auch jetzt schon zum Einsatz kommen und sind nicht davon abhängig, dass ein Anbieter von Funktionen seine Berechnungen umstellt.

Um Bilder in einer für die Privatsphäre schützenden Weise abzurufen, entwickelten Ferreira et al. [FRLD19] eine Methode, die das umsetzen kann. Dafür sollen unterschiedliche Merkmale eines Bildes separiert und einzeln verschlüsselt werden. Sie nennen dafür folgendes Beispiel. Die Farben und Texturen eines Bildes können so separiert werden, dass man auf einem dieser Merkmale Berechnungen ausführen kann und das andere gleichzeitig geschützt bleibt. Dabei wird probabilistisch verschlüsselt. Im Vergleich zu PriTeX werden die Daten also nicht verschleiert, sondern so verschlüsselt, dass sie von Dritten nicht verwendet werden können. Für diese Methode ist allerdings die Kooperation der berechnenden Partei nötig, was für die in PriTeX vorhandenen Techniken nicht nötig ist. Des Weiteren wird in PriTeX auf mehrere Datentypen Rücksicht genommen und nicht nur Bilder geschützt.

Um die Privatsphäre einer Person zu schützen, ist es des Weiteren wichtig, zunächst die genauen Anforderungen der Privatsphäre in einer Art und Weise zu speichern, die auch für Maschinen interpretierbar ist. Dafür können Ontologien [NM01] genutzt werden. Ontologien für die Privatsphäre definieren zunächst die grundlegenden Anforderungen an die Privatsphäre und die daraus folgenden Anforderungen an ein System, welches die Privatsphäre schützen soll. Auch die Relationen dieser Konzepte werden definiert. Es kann daher als eine Art Metadatenmodell angesehen werden, welches die Metadaten der Anforderungen an die Privatsphäre beschreibt. Auch das in dieser Arbeit entwickelte Metadatenmodell in PriTeX beschreibt die grundlegenden Konzepte, sowie die Gleichheiten und Unterschiede der PETs. Daher können die beiden Konzepte verglichen werden. Allgemein lässt sich sagen, dass die Ontologien für die Privatsphäre im Vergleich zu dieser Arbeit an einem anderen Ort ansetzen. Sie bestimmen das Zugriffsrecht von Personen oder Personengruppen auf bestimmte Daten. Die in PriTeX enthaltenen PETs verhindern nicht den Zugriff auf die Daten, sondern sollen stattdessen verhindern, dass Daten auf eine Person zurückführbar sind oder auch die gesammelten Daten über eine Person so zu verändern, dass keine bedeutsamen Rückschlüsse gezogen werden können. Das Metadatenmodell in PriTeX versucht also nicht den Personen eine gute Repräsentation ihrer Anforderungen an die Privatsphäre zu geben, wie es die Ontologien für die Privatsphäre tun. Stattdessen konzentriert es sich darauf, dem Nutzer eine Technik zu geben, die zu seinen Anforderungen passt. Im Folgenden werden zwei solche Ontologien beschrieben.

Dieser Ansatz wird beispielsweise von Arruda und Bulcão-Neto [AB19] umgesetzt. Dort soll eine Ontologie für die Privatsphäre für den Einsatz in Internet of Things (IoT) Systemen entwickelt werden. Sie definieren zunächst Anforderungen der Privatsphäre an das Modell, um aus diesem

im Anschluss die nötige Ontologie zu entwickeln. Dabei soll das, „IoT-Priv“ genannte, Modell leichtgewichtig sein. Das bedeutet, es werden nur die kritischsten Elemente der IoT Domäne repräsentiert. Sie benutzen das Smart Health Szenario, um die Ontologie zu evaluieren. Dazu nennen sie einige Beispiele, wie IoT-Priv verschiedene Zugriffsszenarion beschreiben kann. Im Vergleich zu dieser Ontologie, wird in dieser Arbeit ein anderes Szenario behandelt. Wie im vorherigen Absatz beschrieben, kann diese Ontologie als eine Art Metadatenmodell angesehen werden. Dabei werden die Metadaten der Anforderungen der Nutzer an die Privatsphäre, statt wie im Metadatenmodell die Metadaten der PETs, beschrieben.

Eine andere Ontologie wurde von Gharib und Mylopoulos [GM18] entwickelt. Ihr wurde der Name „COPri“ gegeben. Dabei konzentrieren sich die Autoren auf das Szenario des betreuten Wohnens, also ein anderes Szenario als in dieser Bachelorarbeit. Dabei gehen sie davon aus, dass ein Senior einen Persönlichen digitalen Assistenten in der Form eines Endgerätes besitzt. Dieser kann mit verschiedenen medizinischen Sensoren kommunizieren. Außerdem kann es die gesammelten Informationen an ein Pflegezentrum weitergeben, wo sie zur Überprüfung der Gesundheit und des Wohlbefindens des Seniors verwendet werden kann. Für die Ontologie werden dann verschiedene Konzepte wie Akteure oder Ziele definiert und ihr Verhältnis zueinander beschrieben. Für die Validierung der Ontologie werden Kompetenzfragen gestellt. COOri umfasst einen großen Teil an verschiedenen Anforderungen, Rollen und Gefahren, die im Bereich der Privatsphäre auftreten können. Im Gegensatz dazu betrachtet PriTeX einen Ausschnitt dieser Bereiche. Es behandelt die möglichen Mechanismen für die Privatsphäre. Genauer gesagt, behandelt es mögliche Techniken für den Schutz der Privatsphäre. Auch diese Ontologie kann als Metadatenmodell für die Anforderugnen an die Privatsphäre angesehen werden. Dabei werden in Vergleich zu dem Metadatenmodell in PriTeX andere Metadaten beschrieben.

Ghinita et al. [GKK+08a] behandeln ein ähnliches Problem wie diese Bachelorarbeit. Sie beschreiben die Problematik des Zugriffs auf sensible Daten, die sich aus den zahlreichen unterschiedlichen Lösungen für den Schutz der Privatsphäre, für Endnutzer ergibt. Dabei erklären sie, dass die bisher entwickelten Lösungen sich immer auf einen kleinen Teil der Daten fokussieren, wodurch es für Endnutzer oder Personen ohne Expertise schwierig ist, sich ganzheitlich zu schützen. Um diese Problematik zu lösen, entwickelten die „PRIVACY-AVARE“. Diese konzeptuelle Lösung soll ein verteiltes Privatsphäre-Management ermöglichen. Dabei werden die Präferenzen eines Nutzers einmal zentral definiert und können dann auf allen Geräten umgesetzt werden. Verweigert der Anbieter eines Dienstes die Kooperation, so hat PRIVACY-AVARE drei Möglichkeiten. Es kann leere Daten, verschleierte Daten oder Ersatzdaten senden, um so den Service wiederzubekommen. Grundsätzlich wäre es möglich, dass eines der Geräte, auf denen die Präferenzen umgesetzt werden, ein CV ist. Dabei sollte darauf geachtet werden, dass die Anforderungen an die Funktionalität der Services höher sein kann als bei anderen Geräten. Dies kann zum Beispiel für sicherheitsrelevante Anwendungen der Fall sein. Nach einem Unfall ist es nötig, dass der tatsächliche Standort versendet wird. Es sollte also darauf geachtet werde, in welchen Szenarien die Präferenzen des Nutzers nicht beachtet werden können. PRIVACY-AVARE sieht nicht vor, dass der Nutzer die Technik für die Verschleierung seiner Daten frei auswählt, sondern ermittelt aus den genannten Präferenzen, welche der vorhandenen Techniken verwendet werden sollte. Daher begleitet sie den Nutzer, wie es auch PriTeX machen soll, dabei, seine Daten mit einer geeigneten Technik zu schützen. Ein Nachteil von PRIVACY-AVARE ist es, dass keine Rücksicht darauf genommen wird, dass Nutzer möglicherweise in verschiedenen Szenarien verschiedene Präferenzen für ihre Privatsphäre haben. Auch PriTeX macht diese Unterscheidung nicht. Möchte ein Nutzer zum Beispiel die von seinem Smartphone

aufgezeichneten Ortsdaten stärker schützen, als die seines CV, so sollte das auch dargestellt werden können. Daher ist es wichtig Lösungen anzubieten, die sich genau auf ein Szenario beziehen, so wie es PriTeX für das Szenario der CVs macht.

8 Zusammenfassung und Ausblick

In dieser Arbeit wurde ein System mit dem Namen PriTeX entwickelt. Dieses System soll die Speicherung und Suche nach PETs ermöglichen. Dabei soll es den Nutzern des Systems erleichtert werden, aus den gespeicherten PETs, eines zu wählen, welches die Daten nach seinen Anforderungen schützt. Die in PriTeX gespeicherten PETs sollen dabei Daten schützen, die in einem CV erzeugt werden. Dabei wird auf Softwarelösungen eingegangen, die sich digital speichern lassen. Hier werden vier Datentypen betrachtet. Die Geschwindigkeitsdaten, Ortsdaten, Stimmdaten und Kameradaten.

Für die Speicherung und Verwaltung der PETs und die Suche nach einem für den Nutzer passenden PET aus den gespeicherten PETs, können die Komponenten von PriTeX verwendet werden. Für all diese Vorgänge wird außerdem ein Metadatenmodell genutzt, welches die wesentlichen Eigenschaften der PETs beschrieben. Da das Metadatenmodell den Kern des Systems darstellt, wurde in dieser Arbeit das Augenmerk auf dieses gelegt und für dieses eine Implementierung gegeben. Für PriTeX wurde ein Konzept angegeben, mit welchem sich die Speicherung und Verwaltung von den PETs ermöglicht wird. Durch die Kombination der Komponenten von PriTeX und dem Metadatenmodell wird die Suche ermöglicht.

Zu jedem PET und seinen Implementierungen werden die zugehörigen Metadaten gespeichert. Diese ergeben eine eindeutige Beschreibung der PETs und Implementierungen, durch welche diese voneinander unterschieden und gefunden werden können. Dadurch kann nicht nur ein zu den Anforderungen passendes PET, sondern auch eine passende Implementierung gefunden werden, die im CV des Nutzers ausgeführt werden können.

Insgesamt ermöglichen PriTeX und das zugehörige Metadatenmodell die Suche nach einem passenden PET. Dadurch wird dem Nutzer ermöglicht, seine Privatsphäre, mit der Hilfe eines für ihn passenden PETs, zu schützen.

Ausblick

In dieser Arbeit wurde, wie zuvor beschrieben, das Augenmerk auf das Metadatenmodell gelegt, welches das Kernstück von PriTeX darstellt. Möchte man auch die weiteren Funktionen von PriTeX, abgesehen von der Suche nach PETs, können die Komponenten von PriTeX implementiert werden. Zuvor wurde bereits erwähnt, dass die bisher entwickelten PET oft nicht speziell für den Bereich der CVs konzipiert wurden. Einige von diesen können mit leichten Modifizierungen auch in den CVs eingesetzt werden. Diese Modifizierungen sollten genauer betrachtet und auch durchgeführt werden. Für einige der Metadatenfelder, wie die Ausgabequalität, ist eine Konkretisierung der möglichen Eingaben nötig. Dazu wird auch eine Prüfung der Kompatibilität der Ausgaben der PETs mit den Eingaben der Funktionen nötig. Denn nur aus diesem Zusammenhang lassen sich aussagekräftige Metriken für die Ausgabequalität finden. Des Weiteren kann das erstellte Metadatenmodell erweitert

werden, sodass PETs weiterer Datentypen, wie die Daten eines Bremsvorganges, durch es beschrieben werden können. So kann die Anzahl an PETs in PriTeX erhöht werden. Dadurch können dem Nutzer mehr PETs angeboten werden.

Literaturverzeichnis

[AB19] M. F. Arruda, R. F. Bulcão-Neto. „Toward a Lightweight Ontology for Privacy Protection in IoT“. In: *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*. SAC '19. Limassol, Cyprus: Association for Computing Machinery, 2019, S. 880–888. ISBN: 9781450359337. DOI: 10.1145/3297280.3297367. URL: https://doi.org/10.1145/3297280.3297367 (zitiert auf S. 68).

[ALH+11] S. Amini, J. Lindqvist, J. Hong, J. Lin, E. Toch, N. Sadeh. „Caché: caching location-enhanced content to improve user privacy“. In: Juni 2011, S. 197–210. DOI: 10.1145/1999995.2000015 (zitiert auf S. 40, 41, 57).

[BPV12] J. Becker, W. Probandt, O. Vering. *Grundsätze ordnungsmäßiger Modellierung: Konzeption und Praxisbeispiel für ein effizientes Prozessmanagement*. Springer-Verlag, 2012 (zitiert auf S. 52).

[BS03] A. Beresford, F. Stajano. „Stajano, F.: Location Privacy in Pervasive Computing. IEEE Pervasive Computing 2, 46-55“. In: *Pervasive Computing, IEEE* 2 (Feb. 2003), S. 46–55. DOI: 10.1109/MPRV.2003.1186725 (zitiert auf S. 39, 41).

[BSCB00] M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester. „Image Inpainting“. In: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '00. USA: ACM Press/Addison-Wesley Publishing Co., 2000, S. 417–424. ISBN: 1581132085. DOI: 10.1145/344779.344972. URL: https://doi.org/10.1145/344779.344972 (zitiert auf S. 46).

[BVE+03] J. Borking, P. Verhaar, B. Eck, P. Siepel, G. Blarkom, R. Coolen, M. Uyl, J. Holleman, P. Bison, R. Veer, J. Giezen, A. Patrick, C. Holmes, J. Lubbe, R. Lachman, S. Kenny, R. Song, K. Cartrysse, J. Huizenga, X. Zhou. *Handbook of Privacy and Privacy-Enhancing Technologies The case of Intelligent Software Agents*. Nov. 2003. ISBN: ISBN 90 74087 33 7. DOI: 10.13140/2.1.4888.7688 (zitiert auf S. 20).

[CBYR18] P. Cheng, I. E. Bagci, J. Yan, U. Roedig. „Towards Reactive Acoustic Jamming for Personal Voice Assistants“. In: *Proceedings of the 2nd International Workshop on Multimedia Privacy and Security*. MPS '18. Toronto, Canada: Association for Computing Machinery, 2018, S. 12–17. ISBN: 9781450359887. DOI: 10.1145/3267357.3267359. URL: https://doi.org/10.1145/3267357.3267359 (zitiert auf S. 42, 45, 46).

[CLT+20] Y. Chen, H. Li, S.-Y. Teng, S. Nagels, Z. Li, P. Lopes, B. Y. Zhao, H. Zheng. „Wearable Microphone Jamming“. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. CHI '20. Honolulu, HI, USA: Association for Computing Machinery, 2020, S. 1–12. ISBN: 9781450367080. DOI: 10.1145/3313831.3376304. URL: https://doi.org/10.1145/3313831.3376304 (zitiert auf S. 65).

[CR22] P. Cheng, U. Roedig. „Personal Voice Assistant Security and Privacy—A Survey“. In: *Proceedings of the IEEE* 110.4 (2022), S. 476–507. DOI: 10.1109/JPROC.2022.3153167 (zitiert auf S. 43, 45).

[Deh22] Z. Dehghani. *Data Mesh*. 1. O'Reilly Media, 2022. ISBN: 1492092398,9781492092391 (zitiert auf S. 22).

[DS20] Y. Ding, H. Sato. „Derepo: A Distributed Privacy-Preserving Data Repository with Decentralized Access Control for Smart Health“. In: *2020 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2020 6th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*. 2020, S. 29–35. DOI: 10.1109/CSCloud-EdgeCom49738.2020.00015 (zitiert auf S. 68).

[Eur16] European Parliament and Council of the European Union. *Regulation on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (Data Protection Directive)*. Legislative acts L119. Official Journal of the European Union, Apr. 2016. URL: https://eur-lex.europa.eu/eli/reg/2016/679/oj (zitiert auf S. 66).

[FG20] L. Fan, S. Y. Gunja. „Geopriv4j: An Open Source Repository for Practical Location Privacy“. In: *Proceedings of the Sixth International ACM SIGMOD Workshop on Managing and Mining Enriched Geo-Spatial Data*. GeoRich '20. Portland, Oregon: Association for Computing Machinery, 2020. ISBN: 9781450380355. DOI: 10.1145/3403896.3403968. URL: https://doi.org/10.1145/3403896.3403968 (zitiert auf S. 67).

[Fis09] S. Fischer-Hübner. „Privacy-Enhancing Technologies“. In: *Encyclopedia of Database Systems*. Hrsg. von L. LIU, M. T. ÖZSU. Boston, MA: Springer US, 2009, S. 2142–2147. ISBN: 978-0-387-39940-9. DOI: 10.1007/978-0-387-39940-9_271. URL: https://doi.org/10.1007/978-0-387-39940-9_271 (zitiert auf S. 19).

[FRLD19] B. Ferreira, J. Rodrigues, J. Leitão, H. Domingos. „Practical Privacy-Preserving Content-Based Retrieval in Cloud Image Repositories“. In: *IEEE Transactions on Cloud Computing* 7.3 (2019), S. 784–798. DOI: 10.1109/TCC.2017.2669999 (zitiert auf S. 68).

[GCFB18] C. Gao, V. Chandrasekaran, K. Fawaz, S. Banerjee. „Traversing the Quagmire That is Privacy in Your Smart Home“. In: *Proceedings of the 2018 Workshop on IoT Security and Privacy*. IoT S&P '18. Budapest, Hungary: Association for Computing Machinery, 2018, S. 22–28. ISBN: 9781450359054. DOI: 10.1145/3229565.3229573. URL: https://doi.org/10.1145/3229565.3229573 (zitiert auf S. 43, 45, 46).

[GKK+08a] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, K.-L. Tan. „Private Queries in Location Based Services: Anonymizers Are Not Necessary“. In: *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*. SIGMOD '08. Vancouver, Canada: Association for Computing Machinery, 2008, S. 121–132. ISBN: 9781605581026. DOI: 10.1145/1376616.1376631. URL: https://doi.org/10.1145/1376616.1376631 (zitiert auf S. 69).

[GKK+08b] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, L. Tan. „Private queries in location based services: Anonymizers are not necessary“. In: Juni 2008, S. 121–132. DOI: 10.1145/1376616.1376631 (zitiert auf S. 40, 41).

[GL08] B. Gedik, L. Liu. „Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms“. In: *IEEE Transactions on Mobile Computing* 7.1 (2008), S. 1–18. DOI: 10.1109/TMC.2007.1062 (zitiert auf S. 34).

[GM18] M. Gharib, J. Mylopoulos. „A Core Ontology for Privacy Requirements Engineering“. In: *CoRR* abs/1811.12621 (2018). arXiv: 1811.12621. URL: http://arxiv.org/abs/1811.12621 (zitiert auf S. 69).

[Gre16] S. Grey. „Always On: Privacy Implications of Microphone-Enabled Devices“. In: (2016) (zitiert auf S. 42).

[Hel22] D. Held. „Schutz der Privatsphäre in verteilten Software-Defined-Car-Anwendungen“. Universität Stuttgart, 2022 (zitiert auf S. 37, 38, 41).

[HGXA07] B. Hoh, M. Gruteser, H. Xiong, A. Alrabady. „Preserving Privacy in Gps Traces via Uncertainty-Aware Path Cloaking“. In: *Proceedings of the 14th ACM Conference on Computer and Communications Security*. CCS '07. Alexandria, Virginia, USA: Association for Computing Machinery, 2007, S. 161–171. ISBN: 9781595937032. DOI: 10.1145/1315245.1315266. URL: https://doi.org/10.1145/1315245.1315266 (zitiert auf S. 35, 39, 41).

[JLZ+21] H. Jiang, J. Li, P. Zhao, F. Zeng, Z. Xiao, A. Iyengar. „Location Privacy-preserving Mechanisms in Location-based Services: A Comprehensive Survey“. In: *ACM Computing Surveys* 54 (Jan. 2021), S. 1–36. DOI: 10.1145/3423165 (zitiert auf S. 38).

[LJJ+18] H. Liu, Z. Jie, K. Jayashree, M. Qi, J. Jiang, S. Yan, J. Feng. „Video-Based Person Re-Identification With Accumulative Motion Context“. In: *IEEE Transactions on Circuits and Systems for Video Technology* 28.10 (2018), S. 2788–2802. DOI: 10.1109/TCSVT.2017.2715499 (zitiert auf S. 35).

[NHH07] P. A. NORBERG, D. R. HORNE, D. A. HORNE. „The Privacy Paradox: Personal Information Disclosure Intentions versus Behaviors“. In: *Journal of Consumer Affairs* 41.1 (2007), S. 100–126. DOI: https://doi.org/10.1111/j.1745-6606.2006.00070.x. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1745-6606.2006.00070.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1745-6606.2006.00070.x (zitiert auf S. 66).

[NM01] N. Noy, D. Mcguinness. „Ontology Development 101: A Guide to Creating Your First Ontology“. In: *Knowledge Systems Laboratory* 32 (Jan. 2001) (zitiert auf S. 68).

[NM19] A. Nelus, R. Martin. „Privacy-aware Feature Extraction for Gender Discrimination versus Speaker Identification“. In: *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2019, S. 671–674. DOI: 10.1109/ICASSP.2019.8682394 (zitiert auf S. 44–46).

[NSM05] E. Newton, L. Sweeney, B. Malin. „Preserving privacy by de-identifying face images“. In: *IEEE Transactions on Knowledge and Data Engineering* 17.2 (2005), S. 232–243. DOI: 10.1109/TKDE.2005.32 (zitiert auf S. 48–51).

[OSF17] T. Orekondy, B. Schiele, M. Fritz. „Towards a Visual Privacy Advisor: Understanding and Predicting Privacy Risks in Images“. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017, S. 3706–3715. DOI: 10.1109/ICCV.2017.398 (zitiert auf S. 46).

[PC08] J. K. Paruchuri, S.-c. S. Cheung. „Joint optimization of data hiding and video compression“. In: *2008 IEEE International Symposium on Circuits and Systems*. 2008, S. 3021–3024. DOI: 10.1109/ISCAS.2008.4542094 (zitiert auf S. 49–51).

[PCF15] J. R. Padilla-López, A. A. Chaaraoui, F. Flórez-Revuelta. „Visual privacy protection methods: A survey“. In: *Expert Systems with Applications* 42.9 (2015), S. 4177–4195. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2015.01.041. URL: https://www.sciencedirect.com/science/article/pii/S0957417415000561 (zitiert auf S. 46, 47).

[Pet05] J. Peterson. *A Presence-based GEOPRIV Location Object Format*. RFC 4119. Dez. 2005. DOI: 10.17487/RFC4119. URL: https://www.rfc-editor.org/info/rfc4119 (zitiert auf S. 39, 41).

[PST09] S. N. Patel, J. W. Summet, K. N. Truong. „BlindSpot: Creating Capture-Resistant Spaces“. In: *Protecting Privacy in Video Surveillance*. Hrsg. von A. Senior. London: Springer London, 2009, S. 185–201. ISBN: 978-1-84882-301-3. DOI: 10.1007/978-1-84882-301-3_11. URL: https://doi.org/10.1007/978-1-84882-301-3_11 (zitiert auf S. 47, 50, 51).

[SBS07] G. Schindler, M. Brown, R. Szeliski. „City-Scale Location Recognition“. In: *2007 IEEE Conference on Computer Vision and Pattern Recognition*. 2007, S. 1–7. DOI: 10.1109/CVPR.2007.383150 (zitiert auf S. 46).

[SCJ23] S. Stecklow, W. Cunningham, H. Jin. *Special Report: Tesla workers shared sensitive images recorded by customer cars*. 2023. URL: https://www.reuters.com/technology/tesla-workers-shared-sensitive-images-recorded-by-customer-cars-2023-04-06/ (zitiert auf S. 15).

[SCZ20] K. Sun, C. Chen, X. Zhang. „‚Alexa, Stop Spying on Me!‘: Speech Privacy Protection against Voice Assistants“. In: *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. SenSys ’20. Virtual Event, Japan: Association for Computing Machinery, 2020, S. 298–311. ISBN: 9781450375900. DOI: 10.1145/3384419.3430727. URL: https://doi.org/10.1145/3384419.3430727 (zitiert auf S. 16).

[SG12] R. Sobti, G. Geetha. „Cryptographic hash functions: a review“. In: *International Journal of Computer Science Issues (IJCSI)* 9.2 (2012), S. 461 (zitiert auf S. 51).

[SGB+22] C. Stach, C. Gritti, J. Bräcker, M. Behringer, B. Mitschang. „Protecting Sensitive Data in the Information Age: State of the Art and Future Prospects“. In: *Future Internet* 14.11 (2022). ISSN: 1999-5903. DOI: 10.3390/fi14110302. URL: https://www.mdpi.com/1999-5903/14/11/302 (zitiert auf S. 15, 33).

[Sha10] M. Shashanka. „A Privacy Preserving Framework for Gaussian Mixture Models“. In: *2010 IEEE International Conference on Data Mining Workshops*. 2010, S. 499–506. DOI: 10.1109/ICDMW.2010.109 (zitiert auf S. 48, 50, 51).

[Spi23] Spiegel Gruppe. *Die Geschwindigkeit verrät das Ziel*. 2023. URL: https://www.spiegel.de/wissenschaft/mensch/ueberwachung-per-gps-geschwindigkeit-beim-autofahren-verraet-das-ziel-a-985565.html (zitiert auf S. 36).

[SSS19] M. K. Singh, N. Singh, A. K. Singh. „Speaker's Voice Characteristics and Similarity Measurement using Euclidean Distances“. In: *2019 International Conference on Signal Processing and Communication (ICSC)*. 2019, S. 317–322. DOI: 10.1109/ICSC45622.2019.8938366 (zitiert auf S. 35).

[SV04] J. Schiller, A. Voisard. *Location-based services*. Elsevier, 2004 (zitiert auf S. 38).

[TA10] N. Talukder, S. I. Ahamed. „Preventing Multi-Query Attack in Location-Based Services“. In: *Proceedings of the Third ACM Conference on Wireless Network Security*. WiSec '10. Hoboken, New Jersey, USA: Association for Computing Machinery, 2010, S. 25–36. ISBN: 9781605589237. DOI: 10.1145/1741866.1741873. URL: https://doi.org/10.1145/1741866.1741873 (zitiert auf S. 38).

[TÜV23] TÜV SÜD. *ISO/SAE 21434: Zertifizierung von Cybersicherheit im Automobilbereich*. 2023. URL: https://www.tuvsud.com/de-de/dienstleistungen/produktpruefung-und-produktzertifizierung/zertifizierung-nach-iso-sae-21434 (zitiert auf S. 28, 51).

[Uhl15] E. Uhlemann. „Introducing Connected Vehicles [Connected Vehicles]“. In: *IEEE Vehicular Technology Magazine* 10.1 (2015), S. 23–31. DOI: 10.1109/MVT.2015.2390920 (zitiert auf S. 15).

[UN23] United Nations. *United Nations Guide on Privacy-Enhancing Technologies for Official Statistics*. United Nations Committee of Experts on Big Data and Data Science for Official Statistics. New York 2023. URL: https://unstats.un.org/bigdata (zitiert auf S. 19, 20).

[W3C16] W3C. *Extensible Markup Language (XML)*. 2016. URL: https://www.w3.org/XML/ (zitiert auf S. 55).

[W3C17] W3C. *XQuery 3.1: An XML Query Language*. 2017. URL: https://www.w3.org/TR/xquery-31/ (zitiert auf S. 58).

[WER14] T. Winkler, Á. Erdélyi, B. Rinner. „TrustEYE.M4: Protecting the sensor — Not the camera“. In: *2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. 2014, S. 159–164. DOI: 10.1109/AVSS.2014.6918661 (zitiert auf S. 47, 50, 51).

[WKH23] J. Wieler, T. Kroher, C. Henn. *Daten im Auto: Fluch oder Segen?* 2023. URL: https://www.adac.de/rund-ums-fahrzeug/ausstattung-technik-zubehoer/assistenzsysteme/daten-modernes-auto/ (zitiert auf S. 15).

[XX16] S. Xanthopoulos, S. Xinogalos. „A Review on Location Based Services for Mobile Games“. In: *Proceedings of the 20th Pan-Hellenic Conference on Informatics*. PCI '16. Patras, Greece: Association for Computing Machinery, 2016. ISBN: 9781450347891. DOI: 10.1145/3003733.3003770. URL: https://doi.org/10.1145/3003733.3003770 (zitiert auf S. 38).

[XYC+19] X. Xu, J. Yu, Y. Chen, Y. Zhu, L. Kong, M. Li. „BreathListener: Fine-Grained Breathing Monitoring in Driving Environments Utilizing Acoustic Signals“. In: *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*. MobiSys '19. Seoul, Republic of Korea: Association for Computing Machinery, 2019, S. 54–66. ISBN: 9781450366618. DOI: 10.1145/3307334.3326074. URL: https://doi.org/10.1145/3307334.3326074 (zitiert auf S. 44).

[YJZ+18] D. Yang, K. Jiang, D. Zhao, C. Yu, Z. Cao, S. Xie, Z. Xiao, X. Jiao, S. Wang, K. Zhang. „Intelligent and connected vehicles: Current status and future perspectives“. In: *Science China Technological Sciences* 61 (2018), S. 1446–1471 (zitiert auf S. 15).

[ZLMZ22] X. Zhou, S. Li, L. Ma, W. Zhang. „Driver's attitudes and preferences toward connected vehicle information system“. In: *International Journal of Industrial Ergonomics* 91 (2022), S. 103348. ISSN: 0169-8141. DOI: https://doi.org/10.1016/j.ergon.2022.103348. URL: https://www.sciencedirect.com/science/article/pii/S0169814122000890 (zitiert auf S. 16).

[ZTC12] C. Zhang, Y. Tian, E. Capezuti. „Privacy Preserving Automatic Fall Detection for Elderly Using RGBD Cameras“. In: Juli 2012, S. 625–633. ISBN: 978-3-642-31521-3. DOI: 10.1007/978-3-642-31522-0_95 (zitiert auf S. 48, 50, 51).

[ZTYC16] L. Zhang, S. Tan, J. Yang, Y. Chen. „VoiceLive: A Phoneme Localization Based Liveness Detection for Voice Authentication on Smartphones“. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. CCS '16. Vienna, Austria: Association for Computing Machinery, 2016, S. 1080–1091. ISBN: 9781450341394. DOI: 10.1145/2976749.2978296. URL: https://doi.org/10.1145/2976749.2978296 (zitiert auf S. 44–46).

[ZXS05] Y. Zhang, J. Xiao, M. Shah. „Motion Layer Based Object Removal in Videos“. In: *2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05) - Volume 1*. Bd. 1. 2005, S. 516–521. DOI: 10.1109/ACVMOT.2005.75 (zitiert auf S. 47).

Alle URLs wurden zuletzt am 31. 10. 2023 geprüft.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

Ort, Datum, Unterschrift