STUDY ON THE TECHNICAL EVALUATION OF DE-IDENTIFICATION PROCEDURES FOR PERSONAL DATA IN THE AUTOMOTIVE SECTOR

May 14, 2021

Professur für Mobile Business

& Multilateral Security

Institut für Wirtschaftsinformatik

Prof. Dr. Kai Rannenberg

Dr. Sebastian Pape

Frédéric Tronnier

Sascha Löbner

Campus Westend | Gebäude RuW

Theodor-W.-Adorno-Platz 4

60629 Frankfurt am Main

Telefon +49 (0)69 798 34701

kai.rannenberg@m-chair.de

www.m-chair.de

Acknowledgement

The authors are grateful to the Forschungsvereinigung Automobiltechnik e.V. (FAT e.V.) who funded this research. FAT e.V. is a department of the German Association of the Automotive Industry (German: Verband der Automobilindustrie e. V., VDA).

We are in particular grateful to FAT's working group "AK 31 Elektronik und Software" who not only initiated this research but also contributed to this report by providing input and feedback in various meetings.

1 Table of Contents

2 Executive Summary

The following document represents the final report on the project “Study on the technical evaluation of de-identification procedures for personal data” by Prof. Dr. Rannenberg from the chair of Mobile Business & Multilateral Security at Goethe University Frankfurt am Main, Germany.

The aim of this study was to identify and evaluate different de-identification techniques that may be used in several mobility-related use cases. To do so, four use cases have been defined in accordance with a project partner that focused on the legal aspects of this project, as well as with the VDA/FAT working group. Each use case aims to create different legal and technical issues with regards to the data and information that are to be gathered, used and transferred in the specific scenario. Use cases should therefore differ in the type and frequency of data that is gathered as well as the level of privacy and the speed of computation that is needed for the data.

Upon identifying use cases, a systematic literature review has been performed to identify suitable de-identification techniques to provide data privacy. Additionally, external databases have been considered as data that is expected to be anonymous might be reidentified through the combination of existing data with such external data.

For each case, requirements and possible attack scenarios were created to illustrate where exactly privacy-related issues could occur and how exactly such issues could impact data subjects, data processors or data controllers. Suitable de-identification techniques should be able to withstand these attack scenarios. Based on a series of additional criteria, de-identification techniques are then analyzed for each use case. Possible solutions are then discussed individually in chapters 6.1 – 6.2.

It is evident that no one-size-fits-all approach to protect privacy in the mobility domain exists. While all techniques that are analyzed in detail in this report, e.g., homomorphic encryption, differential privacy, secure multiparty computation and federated learning, are able to successfully protect user privacy in certain instances, their overall effectiveness differs depending on the specifics of each use case.

3 WP1:

The first work package is related to the development of suitable use cases which de-identification techniques are to be evaluated upon.

These use cases were developed and refined in accordance with the FAT working group and the legal chair of Prof. Spindler. The use cases were derived through extensive consultation with the consortium, based on a series of assumptions and restrictions:

- Use cases should differ from each other in their entities and the types of data that are to be used. This was needed in order to derive different legal and technical conclusions from each use case.
- Use cases should be broadly defined and should not be based on actual or planned implementations and use cases of members of the VDA. That is, the use cases should not go into too much detail in order to keep actual services and applications of VDA members confidential.
- Use cases should greatly involve the transfer of data in order to make for interesting case studies from both a legal and a technical perspective.

Based on a series of meetings, four use cases were agreed upon: Electricity provider, Predictive maintenance, Pedestrian in autonomous driving and Social media recommended location

All use cases are explained in the following paragraphs.

3.1 Use Case: Electricity Provider

The electricity provider use case is aimed at deriving insights from multiple in-motion vehicles that are provided to a third party as a service. Here, weather information is collected by vehicles currently driving in an area. This information is then to be aggregated by a business intelligence provider (B-IP) and its results sent to the electricity grid operator (EGO).

Figure 1depicts a high-level overview of the use case while the entities in the use case are described below in more detail.

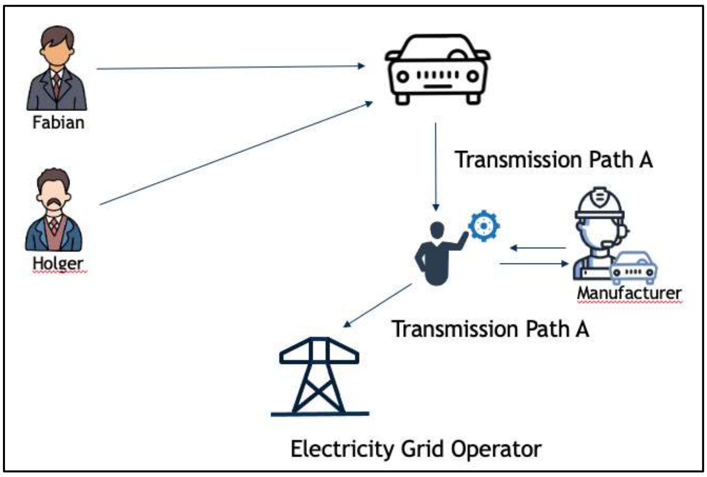


Figure 1. High-level data flow chart of Use Case 1

Entities:

- *Vehicle:* The vehicle driving within a certain geographical area is using multiple sensors to collect live weather information such as brightness, rain and humidity. This information, together with the current vehicle location, is used to provide real-time weather insights as a service. There are multiple vehicles on the road.
- *Vehicle drivers:* Vehicle drivers can be both the owner of a vehicle as well as other individuals such as friends and family members of the vehicle owner. For the technical evaluation of this use case, a differentiation is not necessary.
- *Electricity Grid Operator (EGO):* The EGO is a national electricity provider within Germany. EGO provides electricity through coal and renewable energies such as wind and solar. As the weather can differ across the whole country, exact weather conditions within small regions provide valuable insights on how the overall electricity grid is managed best. Thus, EGO would like to receive exact weather data that is matched to specific regions and locations within the country.
- *Business Intelligence Provider (B-IP):* The B-IP represents a multinational company that is providing data-driven analytics to its customers. Its servers are located abroad. The aim of the B-IP is to gather as much data as possible in order to provide knowledge and insights to the customer.
- *Vehicle Manufacturer:* The vehicle manufacturer is the initiator of the use case and receives information on the correct functioning of the service itself. This may include information on the total amount of data that has been processed and aggregated statistics on the provided service. The vehicle manufacturer is not directly involved in the processing and providing of data.

3.2 Use Case: Pedestrians in Autonomous Driving

Use Case 3 represents the use case where the most sensitive data are processed. Here, an autonomous driving vehicle is collecting data while driving. Pedestrians are standing on the side of the road and are recognized by the vehicle in order to assess the walking direction and speed of the pedestrian. Such data are necessary for the vehicle to evaluate whether a pedestrian will walk onto the road and in the driving path of the vehicle. In this process, data, such as the statue, walking and viewing direction, gait, and facial features of a pedestrian, are processed. In a technical sense, these data are biometric data. Whether such biometric data may be classified as special category data under Art. 9 (1) GDPR, asking for special attention during the processing of it, depends on the interpretation of Art. 9 GDPR – which is still debated. According to the stance of the EDPB these data may only be qualified as data according to Art. 9 (1) GDPR if the data processor has a special intention to process these data. Figure 2 depicts the high-level design of the use case.

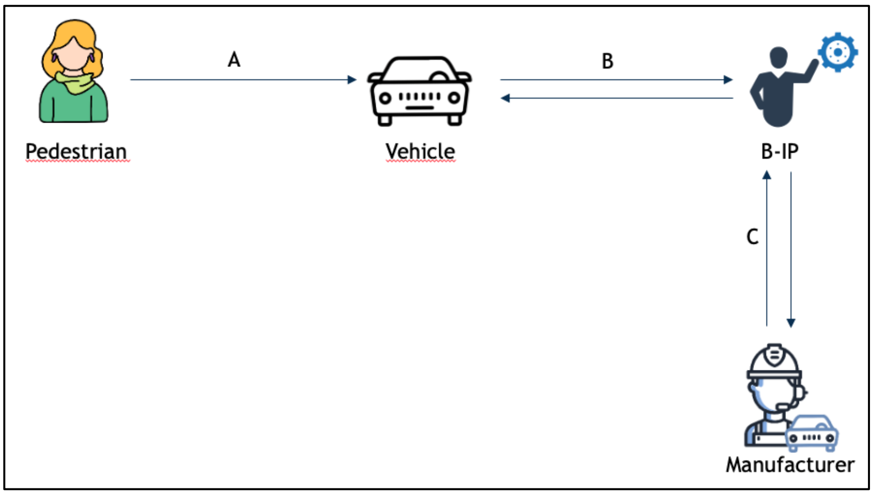


Figure 2. High-level data flow chart of Use Case 3

Entities:

- *Pedestrian:* We define the pedestrian as a random person who is captured by a camera from a vehicle. Whether the vehicle is stationary or driving is neglected. The pedestrian does not want to be identified by the vehicle. Therefore, features that would identify pedestrians must be made unrecognizable in order to anonymize individuals.

- *Vehicle:* For autonomous driving support systems such as automatic braking are required. To evaluate a situation where a pedestrian is moving close to the close to the roadway the direction of movement must be determined. Therefore, the vehicle is equipped with a camera and a machine learning-based model that can predict the direction in which a pedestrian is moving. The vehicle can share information with the B-IP to improve the model.

- *B-IP:* The B-IP is responsible to maintain, train and update the model that is used e.g. for movement prediction. The B-IP can receive vehicle specific information from the manufacturer. The B-IP continuously communicates with the vehicle. The B-IP also creates and sends reports to the manufacturer. These reports contain information about certain vehicle models but not on a certain vehicle.

- *Manufacturer:* The manufacturer has a supporting role and provides vehicle specific information to the B-IP. The manufacturer also receives reports about the status of certain vehicle models.

3.3 Use Case: Social Media Location Recommendation

In-car personalized services are provided by combining internal and external data sources. Car users can access services from social media and platforms in the car. In this example we utilize social media preferences to give recommendations for restaurants. By aggregating information while driving, a personalized restaurant recommendation can be made.

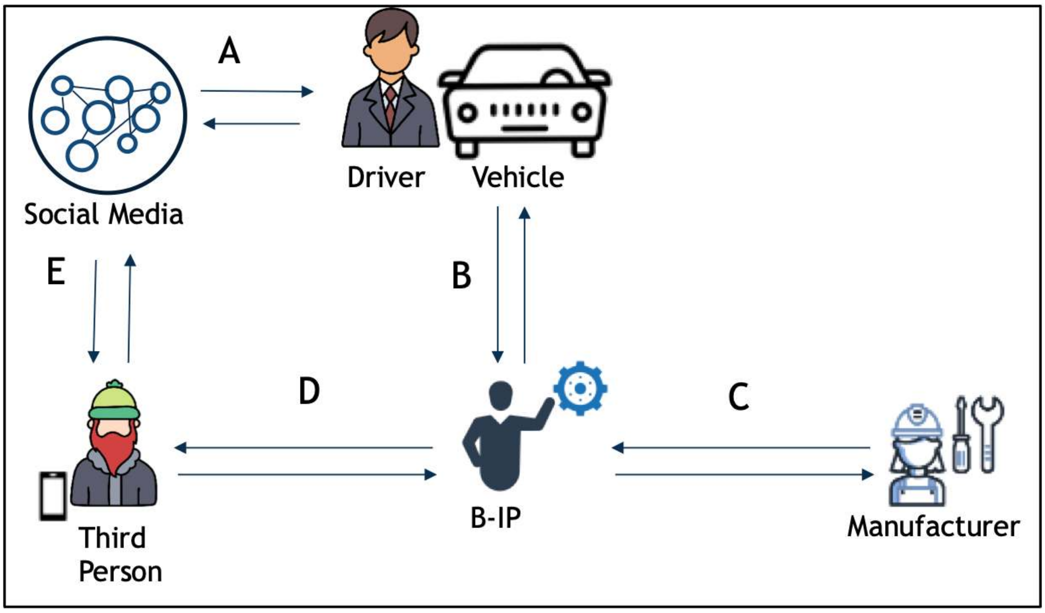


Figure 3 Social Media Services

Driver/ Vehicle: Again, in this use case we treat driver and vehicle as technically one entity. The calculation of a personalized location based on social media preferences is triggered by the driver/vehicle. The driver/vehicle has a communication channel with the social media platform and the B-IP.

Social Media: Information such as recently visited places, food preferences and willingness to pay are stored on a social media platform. The social media platform has a direct communication channel with several users, including the driver and a third person. The social media platform has an indirect communication channel with the B-IP with the driver/vehicle or another device in between. All information that is to be shared on the social media platform has to be authorized by the respective user of the social media platform.

B-IP: The B-IP is responsible for calculating location recommendations such as the restaurant recommendation in this use case. The B-IP indirectly receives information from the social media platform about the users' preferences. Users in this use case are the driver/vehicle and a third person that wants to meet with the driver/vehicle.

Third Person: The third person and the driver/vehicle try to find a restaurant at the intersection of their preferences. Therefore, the third person also authorizes the transmission of social media data between B-IP and the social media platform.

Manufacturer: In this scenario, the manufacturer has no active role. The manufacturer communicates with the B-IP and receives information about model performance and user acceptance.

3.4 Use Case: Predictive Maintenance

Predictive maintenance describes a technique to determine the condition of a machine or specific parts of it to derive the optimal time of maintenance for it. In the case of a vehicle, parts of the vehicle are exchanged and the vehicle is serviced before actual failure occurs. For predictive maintenance, sensors gather information on the status of vehicle parts in order to measure degradation. This provides several advantages for the owner of a vehicle, the vehicle manufacturer as well as the garage that services the vehicle eventually. The vehicle owner is at a lesser risk to break down on the road and can easily plan trips to the garage, while the garage can plan repairs before they occur, making ordering of spare parts easier and improving workload management. For the vehicle manufacturer this means more satisfied customers.

Figure 4 depicts a high-level data flow chart in which the vehicle sends data to the B-IP which in turn notifies the vehicle – and thus the driver – about an upcoming repair.

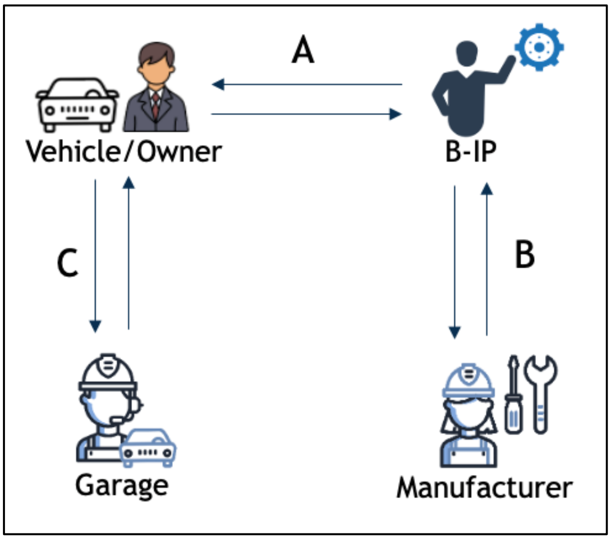


Figure 4. High-level data flow chart for Use Case 2

Entities:

- *Vehicle/Driver:* The vehicle is equipped with many different sensors that constantly collect and store maintenance-related data locally in the car. The vehicle receives a warning from the B-IP if parts are defective or the vehicle needs repair or maintenance. The car can make a repair or maintenance request to the workshop. This must first be approved by the driver. The driver has a communication channel with the garage and can release repair and maintenance orders. Vehicle and driver will be treated as one entity in this scenario, constantly exchanging information.

- *B-IP:* The B-IP takes over the analysis of the vehicle data and has a communication channel with the workshop, the manufacturer and the car.
- *Manufacturer:* The manufacturer is responsible for providing vehicle model-specific information. They also provide information about production defects and recalls. The manufacturer has a communication channel with the B-IP.
- *Garage:* The workshop is responsible for carrying out repairs. Analyses and evaluations that go beyond the actual condition of the vehicle are carried out by the B-IP. The garage receives information from the vehicle about status of the car and the parts that need to be repaired. In case of a defect or a maintenance request, the workshop receives an order from the vehicle.

4 WP2:

The overall objective of WP2 (De-identification and privacy models) has been to create a comprehensive overview on the current status of academic literature on de-identification methods and techniques. To do so, an overview on the models available in ISO/IEC 20889:2018 was combined with techniques that are not mentioned in ISO. For this, several systematic academic literature reviews have been conducted. The aim was to identify "new" de-identification techniques and evaluate whether existing techniques have been greatly improved in the last years in academic papers.

4.1 Literature review procedure

We performed several systematic literature reviews to assess whether their "new" de-identification methods that have not been discussed in ISO exist and were not explicitly stated in the project proposal. The techniques "Secure Multiparty Computation", "Homomorphic Encryption", "Trusted Execution Environment" and "Differential Privacy" were the ones that were already explicitly mentioned in the proposal.

Table 1 provides an overview on the performed searches, exhibiting the database that was searched in, the search term(s), the timeframe and the number of hits and final hits. Here, hits demonstrate the number of findings as given by the search engine while final hits demonstrate the actual findings that were deemed fitting within the context of this project. The literature review focused on the IEEE database as IEEE represents "the world's largest technical professional organization dedicated to advancing technology for the benefit of humanity."[1] The database consists of more than 5 million technical documents from countless academic journals, conferences, transactions and letters, about a third of worldwide technical literature. Additionally, we searched the journal "Proceedings on Privacy-Enhancing Technologies" that focuses solely on PETs for new and innovative solutions in this domain. We focused on more recent findings in order to investigate improvements in established de-identification techniques.

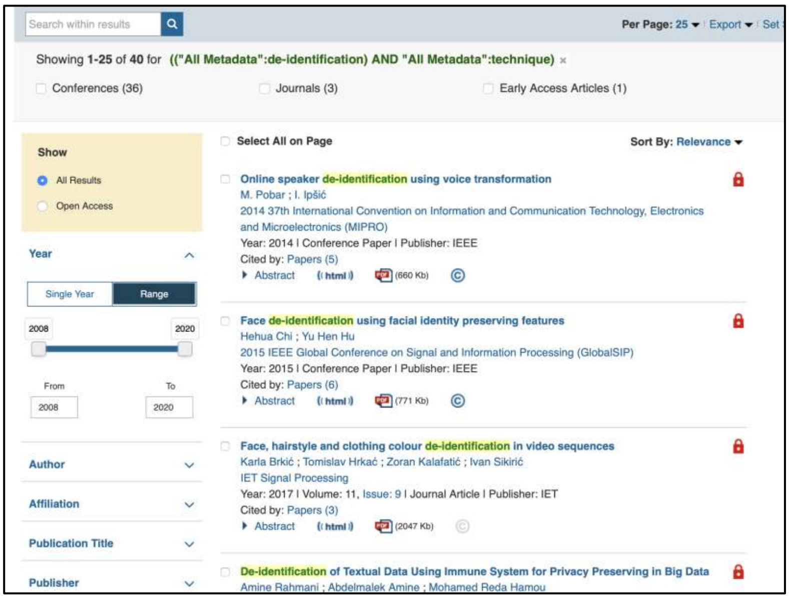
Search	**Search Term**	**From**	**Hits**	**Final Hits**
Proceedings on Privacy-Enhancing Technologies	de-anonymization	2018 - present	24	0
Proceedings on Privacy-Enhancing Technologies	de-identification	2018 - present	19	5

[1] IEEE at a Glance , 2021. Available under: https://www.ieee.org/about/at-a-glance.html (Last accessed: 14.05.2021)

Proceedings on Privacy-Enhancing Technologies	de-identification AND technique	2018 - present	12	0
Proceedings on Privacy-Enhancing Technologies	Trusted Execution Environment	2018 - present	7	5
IEEE Xplore - All Results	de-identification AND technique	2018 - present	40	2
IEEE Xplore - All Results	de-identification AND method	2018 - present	28	0
IEEE Xplore - All Results	data privacy AND method	2018 - present	2518	3
IEEE Xplore - All Results	anonymization AND method	2018 - present	130	4
IEEE Xplore - All Results	data privacy AND technique	2018 - present	644	2
IEEE Xplore - All Results	anonymization AND technique	2018 - present	110	3
IEEE Xplore - All Results	Privacy preservation techniques	2018 - present	165	4
IEEE Xplore - All Results	Secure Multi-Party Computation	2019 - present	99	7
IEEE Xplore - All Results	Differential Privacy AND vehicle	2018 - present	32	7
IEEE Xplore - All Results	Homomorphic Encryption AND vehicle	2018 - present	21	5
IEEE Xplore - All Results	Trusted Execution Environment	2018 - present	20	2

Table 1. List of systematic literature reviews performed

Figure 5 depicts an exemplary result of the literature review search process.



Showing 1-25 of 40 for (("All Metadata":de-identification) AND "All Metadata":technique) ×

□ Conferences (36) □ Journals (3) □ Early Access Articles (1)

● All Results
○ Open Access

2008 2020

From 2008 To 2020

□ Online speaker de-identification using voice transformation
M. Pobar ; I. Ipšić
2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)
Year: 2014 I Conference Paper I Publisher: IEEE
Cited by: Papers (5)
Abstract (html) (660 Kb)

□ Face de-identification using facial identity preserving features
Hehua Chi ; Yu Hen Hu
2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)
Year: 2015 I Conference Paper I Publisher: IEEE
Cited by: Papers (6)
Abstract (html) (771 Kb)

□ Face, hairstyle and clothing colour de-identification in video sequences
Karla Brkić ; Tomislav Hrkać ; Zoran Kalafatić ; Ivan Sikirić
IET Signal Processing
Year: 2017 I Volume: 11, Issue: 9 I Journal Article I Publisher: IET
Cited by: Papers (3)
Abstract (html) (2047 Kb)

□ De-identification of Textual Data Using Immune System for Privacy Preserving in Big Data
Amine Rahmani ; Abdelmalek Amine ; Mohamed Reda Hamou

Figure 5. Exemplary literature review search result

Based on the literature review findings, the following table provides a comprehensive list of de-identification techniques that are to be evaluated in this report. For each technique, a brief description as well as an illustrative example are provided. The table combines de-identification techniques included in ISO/IEC 20889:2018, coded in black, with de-identification techniques that are not included in ISO, coded in red. For every technique, a short description and an illustrative example are provided.

Technique name	**Description**	**Illustrative example or explanation**
Statistical tools		
Sampling	A representative subset of a larger dataset is used for the further processing of data. Various methods to draw a representative subset of data exist.	Of a dataset with 100,000 records, a representative sample of 1000 records will be used for further processing.
Aggregation	A combination of related attributes that provides information at a broader, less detailed level.	Specific address data is aggregated into "City" instead of "Street", "Street Number" and "Postcode" to obtain location information.
Cryptographic tools		
Deterministic encryption	Encryption is a mathematical way to convert information from one form into another by using an external piece of information (the key). Deterministic encryption ensures that, given the same key and input, the ciphertext (output) will always be the same.	The encrypted text: "Hello World" will always result in "Abc Defgh", given the same encryption and key.
Order-preserving encryption	A form of encryption that preserves numerical ordering of the plaintext. This allows for comparison and limited statistical processing of the data while it is still encrypted.	If two values have a fixed ordering in plaintext, the same values will have the same ordering in ciphertext.
Homomorphic encryption	A cryptographic method that allows mathematical (e.g., addition, subtraction) operations on ciphertext	Homomorphic encrypted data is processed, meaning that mathematical operations are

	instead of on non-encrypted plaintext. That means that the result of the processing of the encrypted data matches the result of the processing of the un-encrypted data. This method has the advantage that a third party can process the data without being able to see the actual private data.	performed on the data, without the processor being able to see the plaintext, the non-encrypted data. The results of these operations can then be decrypted only by the holder of the private key.
Homomorphic secret sharing	A secret (information) is homomorphically encrypted and divided into shares that can be distributed to multiple recipients. Only if all or most, of the shares are combined can the secret be decrypted again. If the same mathematical operation is performed on all shares, the result is the one for the original, "full" secret.	A group of people wants to anonymously decide on a topic by voting for or against it. A number of authorities that will later count the votes, provide the voters with a public key. Each voter computes a polynomial through the vote and a random coefficient and sends the result back to the authorities. Authorities collect and calculate the sum of all votes and are then able to determine if more “yes” or “no” counts have been made. No authority knows how a single voter voted, only the aggregate result is broadcast.
Federated learning	Federated learning was first proposed by Google in 2017 with the aim to build a central Machine Learning model based on locally computed submodels. These submodels are trained on a local database and can send/receive updates to/from a central stored model. These updates only contain fragments of the locally stored data and therefore increase privacy protection. In this approach a central database does not exist.	Next-word prediction on smartphones is one of the most common examples where Federated Learning is used nowadays. On each device, a local model is trained that sends updates to the central server from time to time. These updates only contain a pretrained predictor instead of raw user texts. After collecting updates from multiple parties, the central server sends an improved model back. Through this setup the texts written by the user stay secret.
Confidential Computing (Trusted Execution Environment (TEE))	A hardware-based technique to protect and secure the data in use. The data is hereby stored in a so-called Trusted Execution Environment (TEE) in which it cannot be seen or transformed by a debugger. Only	Smartphones or tablets may contain a TEE by manufacturers such as AMD or IMB. In smartphones, TEE can be used for online banking purposes and authenticates

	authorized code can access and operate on the data in the TEE. This environment creates a high degree of trust as threats from “outside” the TEE can be ignored. Other hardware parts are not able to access the TEE.	transactions by managing device drivers such as the fingerprint sensor.
Secure Multiparty Computation	A cryptographic method through which data is secretly shared between multiple parties and processed by a distributed computing protocol that ensure that no information is leaked.	A, B and C want to calculate their average salary without providing the others with their own salary. Each person’s salary is split into three randomly generated shares (e.g., A's salary is 100, three shares of -10, 50. 60 are created). Every person gets 1 share from each person. The average over all shares can now be calculated without any person revealing their true salary.
Suppression		
Masking	Removing all direct identifiers that could identify an individual on its own from the dataset.	Identifiers like "ID" are deleted from the dataset.
Local Suppression	Removing selected values of attributes that in combination with others can identify data principals. This is done to remove "rare values" in a dataset.	If an individual can be identified because it exhibits a unique set of values for several variables, one or more variables can be suppressed in order to prevent de-identification.
Record Suppression	Removing an entire record or records from a dataset.	If an individual can be identified because it exhibits a unique set of values, the whole record of the individual is removed.
Sampling	A representative subset of a larger dataset is used for the further processing of data. Various methods in order to draw a representative subset of data exist.	Of a dataset with 100,000 records, a representative sample of 1000 records will be used for the further processing. This can be seen as the first step, followed by additional de-identification techniques.
Pseudonymization	Creation of pseudonyms that can be independent of the identifying	The identifiers in a dataset are replaced with random or

	attributes, replacing original values of attributes at random, or are derived from identifying attributes using cryptography.	encrypted pseudonyms. This creates additional information such as a list of pseudonyms assigned to actual identifiers or cryptographic keys.
Generalization	Reducing the granularity of information in attributes in a dataset.	Height data of individuals is aggregated into categories: Height in cm is aggregated into the categories small, medium, and large.
Rounding	Rounding numerical values for selected attributes based on a rounding base.	The value 8 gets rounded up to 10 with a probability x and rounded down to 5 with a probability 1-x.
Top/bottom coding	Introducing a threshold for attributes. Values below or above the threshold are replaced with a single value.	The salary category in a survey has a maximum threshold of 100k€.
Randomization	Randomly modifying attributes in a dataset	
Noise addition	Adding random values to the selected attribute with continuous values while still retaining the original statistical properties (mean, variance, correlation...) in the dataset.	Based on a small standard deviation in the dataset, a data record is changed from 5.5 to 5.35.
Permutation	Reordering the values of selected attributes across the records in a dataset. The values are not modified, while the truthfulness of the data is affected, the exact statistical distribution of the selected values across the dataset is retained.	The attribute height and eye-color are randomly changed in a dataset.
Micro aggregation	Replacing all values of continuous attributes with their average.	All values between 0 and 1 are replaced with 0.5, the same is being done for values between 1 and 2 with 1.5.
Differential privacy	A formal privacy measurement model that mathematically guarantees that the result of an analysis on a dataset does not differ stronger than specified, whether a particular data principle is included in the dataset or not. This is done by functions that use noise or dummy data and without changing existing statistical correlations. Several sub-categories	There are two datasets, one of which contains your information. All other data is completely identical. Differential privacy ensures that a query will produce a nearly similar result for both databases.

	exist, e.g., local differential privacy or distributed differential privacy.	
K-anonymity	A privacy measurement model that ensures that the identifying information of each individual is indistinguishable from at least k-1 other individuals in the corresponding equivalence class, making it difficult to link correctly to the associated sensitive attributes.	A database contains multiple entries of vehicle owners and their vehicles. A k-value of 4 now guarantees that for 1 customer, there are at least k=3 other customer from which the 1 customer is indistinguishable for the values in that database.

Table 2. List of de-identification techniques

It can be seen that a wide variety of techniques to protect data and personal data exist. However, not all techniques are suitable in the context of mobility in general and for differing use cases in particular.

In the following, a selected number of findings will be summarized in order to demonstrate the current level of academic research on privacy and de-anonymization.

The total list of findings can be found in Table 3**.** The table provides authors, date and title of the findings as well as an abstract and the main focus of the paper.

Title	Authors	Journal/Conference	Topic	Abstract
A comparative review of privacy preservation techniques in data publishing	Kumar et al., 2018	2018 International Conference on Inventive Systems and Control	De-identification methods	Most enterprises generate a huge amount of public and private dataset actively with the integration of modern technology. So, security is a big concern of these dataset. Initially the security is provided at enterprise level but now-a-days it is an inevitable task to provide security at personal level. So, to achieve the security generalization, suppression, slicing and one attribute per column slicing is used till now. The aim of this paper is to draw a review of all the existing techniques which are used in privacy preservation with comparative analysis of all anonymization techniques and show the flaw of privacy preservation techniques with respect to different parameters.
A Comparative Study of Data Anonymization Techniques	Murthy et al., 2019	2019 IEEE Intl Conference on Big Data Security on Cloud, IEEE Intl Conference on High Performance and Smart Computing and IEEE Intl Conference on Intelligent Data and Security	De-identification methods	In today's digital era, it is a very common practice for organizations to collect data from individual users. The collected data is then stored in multiple databases which contain personally identifiable information (PII). This may lead to a major source of privacy risk for the database. Various privacy preservation techniques have been proposed such as perturbation, anonymization and cryptographic. In this study, five anonymization techniques are compared using the same dataset. In addition to that, this study reviews the strengths and weaknesses of the different technique. In the evaluation of efficiency, suppression is found as the most efficient while swapping is in the last place. It is also revealed that swapping is the most resource-consuming technique while suppressing being less resource consuming.
A Decentralized Location Privacy-Preserving Spatial Crowdsourcing for Internet of Vehicles	Zhang et al., 2020	IEEE Transactions on Intelligent Transportation Systems	Homomorphic Encryption	Abstract—With the rapid development of Internet of Vehicles (IoV), vehicle-based spatial crowdsourcing (SC) applications have been proposed and widely applied to various fields. However, location privacy leakage is a serious issue in spatial crowd- sourcing because workers who participate in a crowdsourcing task are required to upload their driving locations. In this paper, we propose a decentralized location privacy-preserving SC for IoV, which allows vehicle users to securely participate in SC with ensuring the task's location policy privacy and providing multi-level privacy preservation for workers' locations. Specifically, we introduce blockchain technology into SC, which can eliminate the control of vehicle user data by SC-server. We com- bine the additively homomorphic encryption and circle-based location verification to ensure the confidentiality of task's location policy. To achieve multi-level privacy preservation for workers' driving locations, we only reveal a grid where workers are located in. The size of the grid represents the level of privacy preservation. We leverage the order-preserving encryption and non-interactive zero-knowledge proof to prevent workers from illegally obtaining rewards by forging their driving locations. The security analysis results show that our framework can satisfy the above requirements. In addition, the experiment results demonstrate that our framework is efficient and feasible in practice.

A Differential Privacy-Based Protecting Data Preprocessing Method for Big Data Mining	Mo et al, 2019	2019 IEEE International Conference on Trust, Security and Privacy in Computing and Communications/ IEEE International Conference on Big Data Science and Engineering	Differential Privacy	Analyzing clustering results may lead to the privacy disclosure issue in big data mining. In this paper, we put forward a differential privacy-based protecting data preprocessing method for distance-based clustering. Firstly, the data distortion technique differential privacy is used to prevent the distances in distance-based clustering from disclosing the relationships. Differential privacy may affect the clustering results while protecting privacy. Then an adaptive privacy budget parameter adjustment mechanism is applied for keeping the balance between the privacy protection and the clustering results. By solving the maximum and minimum problems, the differential privacy budget parameter can be obtained for different clustering algorithms. Finally, we conduct extensive experiments to evaluate the performance of our proposed method. The results demonstrate that our method can provide privacy protection with precise clustering results.
A journey on privacy protection strategies in big data	Viji et al., 2017	2017 International Conference on Intelligent Computing and Control Systems	De-identification methods	In this modern world providing security for the data is the great challenging task. Especially handling of big data is a great issue because of its volume and variety of data structure. There are various strategies for storing the big data in an efficient way. But the consideration of privacy look up is very important. Privacy preservation varies from different stage of big data life cycle. Due to multi tenancy and massive computation issues, it is become a demanding task. While considering the Framework security, data security, integrity constraints management protecting big data privacy is plays an important role. This paper surveys the privacy requirements, obstacles and the techniques to handle privacy protection strategies in big data.
A study of performance enhancement in big data anonymization	Jang, 2017	2017 International Conference on Computer Applications and Information Processing Technology	K-anonymity	This paper presents the schemes to solve problems when k-anonymity and l-diversity are applied to Big-Data anonymization. The first problem is that information loss and distortion are unavoidable by anonymization job. To reduce the distortion, this paper presents an efficient method that is based on deep anonymization detection. In the method, data publishers analyze the anonymization work, and determine if it is deep or light. If it is thought as deep anonymization, high information distortion is allowed when being distributed to a third party after anonymization. Otherwise, information distortion is kept as low as possible when anonymizing Big-Data to provide the receivers with more meaningful data. The decision for deep anonymization is done by considering a domain data characteristic, data receiver's purpose, and data criticality. The second problem is that it takes much time and requires large buffer space to process the anonymization. To solve the problem, this paper present enhanced read/write schemes.
A Survey on Privacy Preserving Techniques in	Ladole et al., 2018	2018 International Conference on Current	De-identific	Cloud computing methodology is a conceptual based technology which is used widely nowaday. Cloud Computing is an emerging technology which offers an innovative business model for the organizations with massive data without upfront investment, but most of the organizations still hesitate to explore their business over cloud due to security issues. Data privacy protection and data retrieval control is one of the

Cloud Environments		Trends towards Converging Technologies	ation methods	most challenging research works in cloud computing due to confidentiality of user data. Security is one of the major issues which hamper the growth of cloud. This Paper focuses on specific analysis of privacy preservation techniques, comparative analysis and challenges.
A utility preserving data-oriented anonymization method based on data ordering	Salari et al., 2014	2014 International Symposium on Telecommunications	Microaggregation	Due to recent advances, data collection and publishing for scientific purposes are made by some organizations. Published data should be anonymized such that being useful while privacy of data respondents are preserved. So, there is a trade-off between data utility and privacy. Microaggregation is a popular family of anonymization methods that operates on numerical data. In this paper, we propose a microaggregation algorithm called NFPN_MHM that first sorts data in a spiral shape, next it finds a partitioning with the lowest utility loss with respect to the sorted data. Experimental results show that the proposed method attains lower information loss than traditional microaggregation methods and provides a better trade-off between data utility and privacy, especially for scattered data.
Achieving Personalized k-Anonymity-Based Content Privacy for Autonomous Vehicles in CPS	Wang et al., 2019	2020 IEEE Transactions on Industrial Informatics	K-anonymity	Enabled by the industrial Internet, intelligent transportation has made remarkable achievements such as autonomous vehicles by carnegie mellon university (CMU) Navlab, Google Cars, Tesla, etc. Autonomous vehicles benefit, in various aspects, from the cooperation of the industrial Internet and cyber-physical systems. In this process, users in autonomous vehicles submit query contents, such as service interests or user locations, to service providers. However, privacy concerns arise since the query contents are exposed when the users are enjoying the services queried. Existing works on privacy preservation of query contents rely on location perturbation or k-anonymity, and they suffer from insufficient protection of privacy or low query utility incurred by processing multiple queries for a single query content. To achieve sufficient privacy preservation and satisfactory query utility for autonomous vehicles querying services in cyber-physical systems, this article proposes a novel privacy notion of client-based personalized k-anonymity (CPkA). To measure the performance of CPkA, we present a privacy metric and a utility metric, based on which, we formulate two problems to achieve the optimal CPkA in term of privacy and utility. An approach, including two modules, to establish mechanisms which achieve the optimal CPkA is presented. The first module is to build in-group mechanisms for achieving the optimal privacy within each content group. The second module includes linear programming-based methods to compute the optimal grouping strategies. The in-group mechanisms and the grouping strategies are combined to establish optimal CPkA mechanisms, which achieve the optimal privacy or the optimal utility. We employ real-life datasets and synthetic prior distributions to evaluate the CPkA mechanisms established by our approach. The evaluation results illustrate the effectiveness and efficiency of the established mechanisms.

An effective anonymization technique of big data using suppression slicing method	Elanshekhar & Shedge, 2017	2017 International Conference on Energy, Communication, Data Analytics and Soft Computing	Suppression	Now a days there is a large collection of information and is being published in public network. This large data may contain personal information of a person. So, a difficulty in publishing the data of an individual to publish it without the information leak. To avoid the identification of an individual, security must be provided. Many anonymization techniques are used for the privacy of personal information. While publishing the data, techniques like anonymization using generalization and slicing failed to prevent membership disclosure and also has a linkage of information. This eventually led to the loss of utility. Slicing technique uses the horizontal and vertical partitioning for a perfect separation between the uncorrelated attributes to avoid the privacy exploitation. Suppression slicing has overcome this backlogs by comparing the attributes and tuples for similarity check and hide those data values to avoid the linkage and background attack. Thus an effective suppression slicing method is given, which are performed on the attributes having similar values for better utility and privacy.
An Efficient Way of Anonymization Without Subjecting to Attacks Using Secure Matrix Method	Murthy et al., 2018	2018 Second International Conference on Intelligent Computing and Control Systems	Secure Matrix Methods	In current times huge data evolving from multiple sources like hospitals, reservation agencies, online transactions, etc. in massive volumes and obtaining in various forms. These data have privacy concerns due to leakage of data. The outgrowths raised in this situation may drive towards anonymization of sensitive identity information. Let a dataset released for the research purpose by removing the identifying attributes and sensitive attributes, but an adversary find to disclose the identity of the individuals by using the quasi-identifiers and non-sensitive data. Anonymization methods are classified into k-Anonymity, 1-diversity, and t-closeness fail in the better way of hiding the data. These techniques lead to a homogeneous attack, background knowledge attack, and similarity attack. In this article, novel method has been proposed based on secure matrix methods for an effective way of hiding the critical data. This technique accepts the non identified data as an input and produces anonymized data as an output without subjecting to attacks. It experimentally produces better results in anonymizing the data with less execution time.
An Extensive Study on Statistical Data Anonymization Algorithms	Madan & Goswami, 2018	2018 International Conference and Workshops on Recent Advances and Innovations in Engineering	De-identification methods	Gigantic volume of detailed individual information is constantly gathered and divulging of these information is gainful for data mining application. Datais collected from thedata holders by various data publishers before beingthe release of data to the data beneficiary for the purpose of research analysis and mining. This released data may reveal the private and personal information of individuals. Thus arises the most important research issue of privacy in data publishing. Here, in this paper, we provided the analysis of the existing techniques for statistical data anonymization which are used for privacy preservation in published data along with the efficiency and effectiveness of each. This study will help the researchers to understand variety of different anonymization methods for microdata publishing, relationship between k-values, and anonymization degree and execution time.

An Improved method for sharing medical images for Privacy Preserving Machine Learning using Multiparty Computation and Steganography	Vignesh et al., 2019	2019 International Conference on Advances in Computing and Communication	MPC	Digital data privacy is one of the main concerns in today's world. When everything is digitized, there is a threat of private data being misused. Privacy-preserving machine learning is becoming a top research area. For machines to learn, massive data is needed and when it comes to sensitive data, privacy issues arise.With this paper, we combine secure multiparty computation and steganography helping machine learning researchers to make use of a huge volume of medical images with hospitals without compromising patients' privacy. This also has application in digital image authentication. Steganography is one way of securing digital image data by secretly embedding the data in the image without creating visually perceptible changes. Secret sharing schemes have gained popularity in the last few years and research has been done on numerous aspects.
Anonymization Techniques for Protecting Privacy: A Survey	Pawar et al., 2018	2018 IEEE Punecon	De-identification methods	Anonymization is one of fruitful privacy protection technique used in various technology fields such as data mining, cloud computing, big data to secure very sensitive data against third party. In today's world, the value and the amount of data is increasing, hence the protection of data against all possible threats are equally necessary. This paper focuses a brief on data anonymization and differential privacy techniques. Various anonymization techniques which are researched by various researchers across various fields have limitations such as communication and computation cost overhead, accuracy of results after data Anonymization and possibility of different types of attacks. The paper discussed all these issues and their counter-measures through readings of various papers. Finally, this paper presents detailed discussion about existing anonymization techniques (Data anonymization and differential privacy), their comparative analysis by leaving a footprints of future research directions.
AnonymousNet: Natural Face De-Identification with Measurable Privacy	Li & Lin, 2019	2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops	Biometric data de-identification	With billions of personal images being generated from social media and cameras of all sorts on a daily basis, security and privacy are unprecedentedly challenged. Although extensive attempts have been made, existing face image de-identification techniques are either insufficient in photo-reality or incapable of balancing privacy and usability qualitatively and quantitatively, i.e., they fail to answer counterfactual questions such as "is it private now?", "how private is it?", and "can it be more private?" In this paper, we propose a novel framework called AnonymousNet, with an effort to address these issues systematically, balance usability, and enhance privacy in a natural and measurable manner. The framework encompasses four stages: facial attribute estimation, privacy-metric-oriented face obfuscation, directed natural image synthesis, and adversarial perturbation. Not only do we achieve the state-of-the-arts in terms of image quality and attribute prediction accuracy, we are also the first to show that facial privacy is measurable, can

				be factorized, and accordingly be manipulated in a photo-realistic fashion to fulfill different requirements and application scenarios. Experiments further demonstrate the effectiveness of the proposed framework.
Blockchain Applications with Privacy using Efficient Multiparty Computation Protocols	Innocent & Prakash, 2019	2019 PhD Colloquium on Ethically Driven Innovation and Technology for Society	MPC	Blockchain technology provides a distributed solution, but not privacy of data used. Data privacy is included with the help of secure multiparty computation protocols and which in turn increases the complexity of application. This paper provides an efficient solution for blockchain technology with privacy by including a novel optimization for secure computation protocols.
Cryptograhy and Pk-Anonymization Methods for Secure Data Storage in Cloud	Shaik et al., 2019	2019 Third International conference on I-SMAC	K-anonymity	Cloud computing empowers the clients to get outsourced information from cloud storage with no equipment and programming administrations. For compelling the usage of secret information from Cloud Service Provider (CSP), the information owner encodes before storing in the cloud. To secure information in cloud, information protection is a testing assignment. In order to deal this issue, a proficient information security strategy utilizing cryptographic procedures and Pk-Anonymization method is introduced. Accordingly, the proposed technique scrambles the sensitive information, as well as distinguishes the exploitative gathering to get to the information utilizing consolidated hash capacities. Anonymization is fundamental concept for bringing about assurance on users secret information. Protection of Data is likely necessary because of the control lessening system and Pk-Anonymization method. Pk-Anonymization and cryptography in Cloud computing to improve data security is introduced in the proposed method.
Data Querying and Access Control for Secure Multiparty Computation	Maltitz et al., 2019	2019 IFIP/IEEE International Symposium on Integrated Network Management	MPC	In the Internet of Things and smart environments data, collected from distributed sensors, is typically stored and processed by a central middleware. This allows applications to query the data they need for providing further services. However, centralization of data causes several privacy threats: The middleware becomes a third party which has to be trusted, linkage and correlation of data from different context becomes possible and data subject lose control over their data. Hence, other approaches than centralized processing should be considered. Here, Secure Multiparty Computation is a promising candidate for secure and privacy-preserving computation happening close to the sources of the data. In order to make SMC fit for application in these contexts, we extend SMC to act as a service: We provide elements which allow third parties to query computed data from a group of peers performing SMC. Furthermore, we establish fine-granular access control on the level of individual data queries, yielding data protection of the

				computed results. By adding measures to inform data sources about requests and the usage of their data, we show how a fully privacy-preserving service can be built on the foundation of SMC.
Efficient Privacy-Preserving Scheme for Location Based Services in VANET System	Farouk et al., 2020	IEEE Access	Homo-morphic Encryption	A Vehicular Ad-hoc Network (VANET) is a type of Mobile Ad-hoc Network (MANET) that is used to provide communications between nearby vehicles, and between vehicles and fixed infrastructure on the roadside. VANET is not only used for road safety and driving comfort but also for infotainment. Communication messages in VANET can be used to locate and track vehicles. Tracking can be beneficial for vehicle navigation using Location Based Services (LBS). However, it can lead to threats on location privacy of vehicle users; since it can profile them and track their physical location. Therefore, to successfully deploy LBS, user's privacy is one of major challenges that must be addressed. In this paper, we propose Privacy-Preserving Fully Homomorphic Encryption over Advanced Encryption Standard (P2FHE-AES) scheme for LBS query. This scheme is required for location privacy protection to encourage drivers to use this service without any risk of being pursued. It is implemented using Network Simulator (NS-2), Simulation of Urban Mobility (SUMO), and Cloud simulation (CloudSim). Analysis and evaluation results demonstrate that P2FHE-AES scheme can preserve the privacy of the drivers' future routes in an efficient and secure way. The results prove the feasibility and efficiency of P2FHE-AES scheme in terms of query's response time, query accuracy, throughput and query overhead.

Energy-Efficient End-to-End Security for Software Defined Vehicular Networks	Raja et al., 2020	IEEE Transactions on Industrial Informatics	Homo-morphic Encryption	One of the most promising application area of Industrial Internet of Things (IIoT) is Vehicular Ad hoc NETworks (VANETs). VANETs are largely used by Intelligent Transportation Systems (ITS) to provide smart and safe road transport. To reduce the network burden, Software Defined Networks (SDNs) acts as a remote controller. Motivated by the need for greener IIoT solutions, this paper proposes an energy-efficient end-to-end security solution for Software Defined Vehicular Networks (SDVN). Besides SDN's flexible network management, network performance, and energy-efficient end-to-end security scheme plays a significant role in providing green IIoT services. Thus, the proposed SDVN provides lightweight end-to-end security. The end-to-end security objective is handled in two levels: i) In RSU-based Group Authentication (RGA) scheme, each vehicle in the RSU range receives a group id-key pair for secure communication and ii) In private-Collaborative Intrusion Detection System (p-CIDS), SDVN detects the potential intrusions inside the VANET architecture using collaborative learning that guarantees privacy through a fusion of differential privacy and homomorphic encryption schemes. The SDVN is simulated using NS2 & Matlab, and the simulation results provide higher energy efficiency through reduced end-to-end security communication cost and decentralized learning compared with other existing mechanisms. In addition, the p-CIDS detects the intruder with an accuracy of 96.81% in the SDVN.
Improved Strongly Deniable Authenticated Key Exchanges for Secure Messaging	Unger & Goldberg, 2018	2018 Proceedings on Privacy Enhancing Technologies	Key exchanges	A deniable authenticated key exchange (DAKE) protocol establishes a secure channel without producing cryptographic evidence of communication. A DAKE offers strong deniability if transcripts provide no evidence even if long-term key material is compromised (offline deniability) and no outsider can obtain evidence even when interactively colluding with an insider (online deniability). Unfortunately, existing strongly deniable DAKEs have not been adopted by secure messaging tools due to security and deployability weaknesses. In this work, we propose three new strongly deniable key exchange protocols—DAKEZ, ZDH, and XZDH—that are designed to be used in modern secure messaging applications while eliminating the weaknesses of previous approaches. DAKEZ offers strong deniability in synchronous network environments, while ZDH and XZDH can be used to construct asynchronous secure messaging systems with offline and partial online deniability. DAKEZ and XZDH provide forward secrecy against active adversaries, and all three protocols can provide forward secrecy against future quantum adversaries while remaining classically secure if attacks against quantum-resistant cryptosystems are found. We seek to reduce barriers to adoption by describing our protocols from a practitioner's perspective, including complete algebraic specifications, cryptographic primitive recommendations, and prototype implementations. We evaluate concrete instantiations of our DAKEs and show that they are the most efficient strongly deniable schemes; with all of our classical security guarantees, our exchanges require

				only 1 ms of CPU time on a typical desktop computer and at most 464 bytes of data transmission. Our constructions are nearly as efficient as key exchanges with weaker deniability, such as the ones used by the popular OTR and Signal protocols.
Information Entropy Differential Privacy: A Differential Privacy Protection Data Method Based on Rough Set Theory	Li et al., 2019	2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress	Differential Privacy	Data have become an important asset for analysis and behavioral prediction, especially correlations between data. Privacy protection has aroused academic and social concern given the amount of personal sensitive information involved in data. However, existing works assume that the records are independent of each other, which is unsuitable for associated data. Many studies either fail to achieve privacy protection or lead to excessive loss of information while applying data correlations. Differential privacy, which achieves privacy protection by injecting random noise into the statistical results given the correlation, will improve the background knowledge of adversaries. Therefore, this paper proposes an information entropy differential privacy solution for correlation data privacy issues based on rough set theory. Under the solution, we use rough set theory to measure the degree of association between attributes and use information entropy to quantify the sensitivity of the attribute. The information entropy difference privacy is achieved by clustering based on the correlation and adding personalized noise to each cluster while preserving the correlations between data. Experiments show that our algorithm can effectively preserve the correlation between the attributes while protecting privacy.
Introducing Differential Privacy to the Automotive Domain: Opportunities and Challenges	Nelson & Olovsson, 2017	IEEE Vehicular Technology Conference	Differential Privacy	For vehicular data, differential privacy can be especially tricky to enforce due to the fact that vehicles contain a system of thousands of dependent signals collected over time. Consequently, the automotive domain is very complex from a privacy perspective. However, as differential privacy is the only privacy model that provides provable privacy guarantees, this is currently the only robust way of mitigating re-identification attacks on data while maintaining utility. Thus, we believe that the automotive industry will benefit from carrying out their privacy-preserving analyses under differential privacy. In order to properly implement differential privacy, it is vital that the company first model the privacy within their domain, to determine what they are trying to protect. From the model

Location Privacy Protection Based on Differential Privacy Strategy for Big Data in Industrial Internet of Things	Yin et al., 2017	2018 IEEE Transactions on Industrial Informatics	Differential Privacy	In the research of location privacy protection, the existing methods are mostly based on the traditional anonymization, fuzzy and cryptography technology, and little success in the big data environment, for example, the sensor networks contain sensitive information, which is compulsory to be appropriately protected. Current trends, such as "Industrie 4.0" and Internet of Things (IoT), generate, process, and exchange vast amounts of security-critical and privacy-sensitive data, which makes them attractive targets of attacks. However, previous methods overlooked the privacy protection issue, leading to privacy violation. In this paper, we propose a location privacy protection method that satisfies differential privacy constraint to protect location data privacy and maximizes the utility of data and algorithm in Industrial IoT. In view of the high value and low density of location data, we combine the utility with the privacy and build a multilevel location information tree model. Furthermore, the index mechanism of differential privacy is used to select data according to the tree node accessing frequency. Finally, the Laplace scheme is used to add noises to accessing frequency of the selecting data. As is shown in the theoretical analysis and the experimental results, the proposed strategy can achieve significant improvements in terms of security, privacy, and applicability.
Olympus: Sensor Privacy through Utility Aware Obfuscation	Raval et al., 2018	2019 Proceedings on Privacy Enhancing Technologies	Obfuscation	Personal data garnered from various sensors are often offloaded by applications to the cloud for analytics. This leads to a potential risk of disclosing private user information. We observe that the analytics run on the cloud are often limited to a machine learning model such as predicting a user's activity using an activity classifier. We present Olympus, a privacy framework that limits the risk of disclosing private user information by obfuscating sensor data while minimally affecting the functionality the data are intended for. Olympus achieves privacy by designing a utility aware obfuscation mechanism, where privacy and utility requirements are modeled as adversarial networks. By rigorous and comprehensive evaluation on a real world app and on benchmark datasets, we show that Olympus successfully limits the disclosure of private information without significantly affecting functionality of the application.

PAPU: Pseudonym Swap with Provable Unlinkability Based on Differential Privacy in VANETs	Li et al., 2020	2020 IEEE Internet of Things Journal	Differential Privacy	Nowadays, the pseudonym swap has become the mainstream technology for protecting vehicles' trajectory privacy in vehicle ad hoc networks. However, the existing pseudonym swap methods cannot strictly provide the unlinkability between the new pseudonym and old pseudonym of the vehicle due to the lack of theoretical privacy guarantee, resulting in severe leakages of vehicles' trajectory privacy. Our experiment also proves this point and we find that existing works may cause vehicle's pseudonyms to be linked with a probability higher than 60% because they always choose two vehicles with very different driving states (e.g., speeds, directions, and positions) to swap their pseudonyms. To solve this issue, we first give a formal privacy definition based on generalized differential privacy, called pseudonym indistinguishability, to provide a strict unlinkability for pseudonym swap. Then, we design an appropriate utility metric and a new pseudonym swap mechanism, which selects a pseudonym for a vehicle by adapting a differential privacy exponential mechanism to satisfy pseudonym indistinguishability. Abstracting from attackers' prior knowledge, we can strictly guarantee that if two vehicles have a high similarity of driving states, it is impossible for attackers to link the vehicles and their pseudonyms after the swap. Theoretical analyses prove that our mechanism satisfies the proposed privacy definition, thus ensuring the unlinkability between the new pseudonym and the old pseudonym. Extensive experiments on a real data set show that our work only requires about 50% of pseudonym quantities compared to other works and can make the vehicle successfully complete the swap process with a probability of more than 90%, which is higher than any of existing works.
Performance Impact Analysis of Rounds and Amounts of Communication in Secure Multiparty Computation Based on Secret Sharing	Fălămaş & Márton, 2019	2019 RoEduNet Conference: Networking in Education and Research	MPC	A somewhat similar performance evaluation was carried out in [6] for two different implementations of the comparison algorithm: based on homomorphic encryption and based on Shamir secret sharing. The experimental results showed that the implementation of MPC using secret sharing outperforms the implementation using homomorphic encryption considering time efficiency, especially for many computing parties (tests with up to 60 parties were performed).

Persistent Transportation Traffic Volume Estimation with Differential Privacy	Yang et at., 2019	2019 IEEE SmartWorld	Differential Privacy	Traffic volume estimation is critical to the transportation engineering. Persistent traffic volume reveals the amount of core, stable traffic at locations of interest, which is meaningful to many transportation applications, such as traffic flow guidance system. Unfortunately, most of the existing state- of-the-art studies that concentrate on the persistent traffic estimation issue only provide limited privacy preservation. To tackle this challenge, we present two estimators with differential privacy respectively for estimating the persistent point traffic volume and the persistent common traffic volume in this work. We first encode the passing vehicles in privacy-preserving data structures by using the random communications between vehicles and Road-Side Units (RSUs). Then, we derive the persistent traffic estimators through mathematical analysis and bitwise operations. We also prove that the proposed schemes can achieve the ε-differential privacy for protecting the location and trajectory privacy of vehicles through rigorous theoretical analysis. The experimental results based on the real transportation traffic traces data demonstrate the effectiveness of the proposed estimators.
Preserving Privacy in the Internet of Connected Vehicles	Ghane et al., 2020	IEEE Transactions on Intelligent Transportation Systems	Differential Privacy	Today's vehicles are advancing from stand-alone transportation means to vehicle-to-vehicle, and vehicle-to- infrastructure communications enabled devices which are able to exchange data through the transportation communication infrastructure. As the IoT and data remain intrinsically linked together, the fast-changing mobility landscape of intent-based networking for the Internet of connected vehicles comes with a great risk of data security and privacy violations. This paper considers the privacy issues in the distributed edge computing, in which the data is communicated between a number of vehicles in the IoT layer and potentially untrusted edge controllers at the edge of the network. The sensory data communicated by the vehicles contain sensitive information, such as location and speed, which could violate the users' privacy if they are leaked with no perturbation. Recent studies suggest mechanisms for randomizing the stream of data to ensure individuals' privacy. Although the past works on differential privacy provide a strong privacy guarantee, they are limited to applications where communication parties are trusted and/or there is no correlation between the users or the featured of sensory data. In this paper, we address this gap by proposing a differentially private data streaming system that adds a correlated noise in the vehicle's side (IoT layer) rather than the transportation infrastructure. Also, our system is able to ensure a strong privacy level over time. The proposed mechanism is data-adaptive and scales the noise with respect to the data correlation. Our extensive experiments demonstrate that the utility of the output generated by our method outperforms the recent approaches.
Privacy Preserving Big Data Publication On Cloud Using	Andrew et al., 2019	2019 International Conference on Advanced	K-anonymity	In recent trends, privacy preservation is the most predominant factor, on big data analytics and cloud computing. Every organization collects personal data from the users actively or passively. Publishing this data for research and other analytics without removing Personally Identifiable Information (PII) will lead to the privacy breach. Existing anonymization techniques are failing to maintain the balance between data

Mondrian Anonymization Techniques and Deep Neural Networks		Computing & Communication Systems		privacy and data utility. In order to provide a trade-off between the privacy of the users and data utility, a Mondrian based k-anonymity approach is proposed. To protect the privacy of high-dimensional data Deep Neural Network (DNN) based framework is proposed. The experimental result shows that the proposed approach mitigates the information loss of the data without compromising privacy.
Privacy Preserving Deep Learning using Secure Multiparty Computation	Sayyad, 2020	2020 Second International Conference on Inventive Research in Computing Applications	MPC	Many different types of problems have been solved using deep learning in recent pasts. Deep learning techniques are useful for finding solutions to different types of data type's right from structures to semi structures or unstructured. Problems that are based on clustering, classification regression are effectively implemented using deep learning techniques. This utility of machine and deep learning techniques calls for keeping these services on cloud. Providing machine learning as a service to cloud opens problems of security concern of the data involved in training that belongs to different parties involved in training and also the security concerns arises for the data model being trained. This paper has implemented a privacy preserving technique based on secure multi-party computation that creates secret shared to solve the privacy issues for the data involved in training. Our experimental analysis is carried out using MNIST dataset for hand written character recognition as data for learning problem. Experimental analysis indicated that MNIST dataset can be trained to better accuracy using secure multiparty computation and keep the data secured on the network. The PyTorch and PySyft libraries are used for experimentation.
Real-Time Privacy-Preserving Data Release Over Vehicle Trajectory	Ma et al. 2019	2019 IEEE Transactions on Vehicular Technology	Differential Privacy	Intelligent connected vehicle trajectory data are of great value for data mining applications such as traffic management and commercial institutions. However, the leakage of sensitive trajectory makes the user hesitate to use the system if no privacy-preserving mechanism is adopted. In this paper, we propose a privacy-preserving mechanism with differential privacy called RPTR, which protects a vehicle's real-time trajectory data release. First, RPTR adopts a dynamic sampling method to process the trajectory data to meet the application load and practicability. Meanwhile, to ensure the data availability, ensemble Kalman filter based on users' position transfer probability matrix is used in the prediction calculation. Also, we construct the privacy budget allocation method based on regional privacy weight to provide better protection for regions with high user density. Through our analysis and experiments, RPTR not only protects the privacy of real-time trajectory data but also guarantees the data availability.

Ridra: A Rigorous Decentralized Randomized Authentication in VANETs	Sun et al., 2018	IEEE Access	Homomorphic Encryption	Ensuring the security and privacy of vehicle is one of the critical requirements for the safety and reliability of vehicular ad hoc networks (VANETs). A variety of (conditional) anonymous authentication schemes, including group/ring signatures, pseudo-identity-based and PKI-based approaches, have been proposed to achieve highly effective privacy-preserving authentications. A recent effort, i.e., randomized authentication, leverages homomorphic encryption for vehicles to self-generate authenticated identities to achieve full anonymity. Notwithstanding a very attractive feature to prevent single-party traceability, randomized authentication faces a great challenge on the centralized data updating and the frequent clock synchronizations. It also fails to meet the necessity of non-repudiation. In this paper, we present a rigorous decentralized randomized authentication framework with conditional privacy preservation. We use homomorphic encryption and a one-way hash chain for a vehicle to self-generate randomized pseudo-identities. We deploy the pseudonym validation mechanism over the roadside units in order to support decentralized mutual identity authentication and ownership validation of vehicles, in a loosely-coupled or a compound manner. Our framework can provide rigorous Level 3 privacy and traceability of vehicles. We also provide a security condition on valid random values to ensure the uniqueness of pseudonym and non-repudiation of vehicles. The performance evaluation shows that our framework is generally more efficient on infrastructures, in terms of computational and communication overheads than the state-of-art randomized authentications.
RON-Gauss: Enhancing Utility in Non-Interactive Private Data Release	Chanyaswad et al., 2018	2019 Proceedings on Privacy Enhancing Technologies	Differential Privacy	A key challenge facing the design of differential privacy in the non-interactive setting is to maintain the utility of the released data. To overcome this challenge, we utilize the Diaconis-Freedman-Meckes (DFM) effect, which states that most projections of high-dimensional data are nearly Gaussian. Hence, we propose the RON-Gauss model that leverages the novel combination of dimensionality reduction via random orthonormal (RON) projection and the Gaussian generative model for synthesizing differentially-private data. We analyze how RON-Gauss benefits from the DFM effect, and present multiple algorithms for a range of machine learning applications, including both unsupervised and supervised learning. Furthermore, we rigorously prove that (a) our algorithms satisfy the strong ε-differential privacy guarantee, and (b) RON projection can lower the level of perturbation required for differential privacy. Finally, we illustrate the effectiveness of RON-Gauss under three common machine learning applications – clustering, classification, and regression – on three large real-world datasets. Our empirical results show that (a) RON-Gauss outperforms previous approaches by up to an order of magnitude, and (b) loss in utility compared to the non-private real data is small. Thus, RON-Gauss can serve as a key enabler for real-world deployment of privacy-preserving data release.

SafePub: A Truthful Data Anonymization Algorithm With Strong Privacy Guarantees	Bild et al., 2018	2018 Proceedings on Privacy Enhancing Technologies	Differential Privacy	Methods for privacy-preserving data publishing and analysis trade off privacy risks for individuals against the quality of output data. In this article, we present a data publishing algorithm that satisfies the differential privacy model. The transformations performed are truthful, which means that the algorithm does not perturb input data or generate synthetic output data. Instead, records are randomly drawn from the input dataset and the uniqueness of their features is reduced. This also offers an intuitive notion of privacy protection. Moreover, the approach is generic, as it can be parameterized with different objective functions to optimize its output towards different applications. We show this by integrating six well-known data quality models. We present an extensive analytical and experimental evaluation and a comparison with prior work. The results show that our algorithm is the first practical implementation of the described approach and that it can be used with reasonable privacy parameters resulting in high degrees of protection. Moreover, when parameterizing the generic method with an objective function quantifying the suitability of data for building statistical classifiers, we measured prediction accuracies that compare very well with results obtained using state-of-the-art differentially private classification algorithms.
Secure Multiparty Computation via Homomorphic Encryption Library	Ghanem & Moursy, 2019	2019 International Conference on Intelligent Computing and Information Systems	MPC	Secure multiparty computation (MPC) is required when individuals want to privately evaluate a function over their inputs. While evaluating a common function, the participants do not reveal their inputs to each other. A homomorphic encryption (HE) scheme allows the evaluation of arbitrary computations on encrypted data without decrypting it. In theory, realizing MPC through a HE scheme is a simple and efficient approach. However, despite its promising theoretical power, the practical side of the approach remains underdeveloped. In this work, motivated by the rising MPC applications, e.g. cloud computation, a HE library is extended to provide the necessary methods for MPC. In particular HElib that implements Brakerski-Gentry-Vaikuntanathan (BGV), a HE scheme, is extended to support MPC protocols. This extension provides a broadcast protocol for the generation of a global public key by N parties, where each party maintains a share of the corresponding private key. In addition, the homomorphic evaluation of functions on ciphertexts encrypted by the public key is extended. Furthermore, a decryption broadcast protocol is provided where ciphertexts are decrypted using the individual shares of the private key. The proposed extension can be adapted to other HE libraries. A second contribution of this work, is a 2 n factorial experimental design and analysis to study the memory, computation, and communication costs of HElib and the proposed extension. Four main factors are identified: the security parameter, the plaintext space, the number of levels of the evaluation function, and the number of parties. The proposed extensions are shown to be effective and efficient. On the experimented setup, it takes about 0.2 sec for multiparty key generation and 0.06 sec for multiparty decryption.

Simulation-based evaluation of techniques for privacy protection in VANETs	Tomandl et al., 2012	2012 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications	K-anonymity	In vehicular ad hoc networks (VANETs) tracking of participants is an issue that is examined by many research groups. These groups came up with several different concepts of counter measures against tracking attacks. All of these presented techniques seem to offer a pretty good protection. We pick out two very promising concepts - the Mix Zones and the Silent Periods - to examine them in a simulation environment to actually identify their strengths and weaknesses. Our simulation results show rather high success rates for attackers with relatively unsophisticated attack heuristics. Furthermore we confirm the correlation between several influencing factors and the success rates of attacks and study the connection to the common metrics k-anonymity and entropy.
SoK: Differential Privacy as a Causal Property	Tschantz et al., 2020	2020 IEEE Symposium on Security an Privacy	Differential Privacy	We present formal models of the associative and causal views of differential privacy. Under the associative view, the possibility of dependencies between data points precludes a simple statement of differential privacy's guarantee as conditioning upon a single changed data point. However, we show that a simple characterization of differential privacy as limiting the effect of a single data point does exist under the causal view, without independence assumptions about data points. We believe this characterization resolves disagreement and confusion in prior work about the consequences of differential privacy. The associative view needing assumptions boils down to the contrapositive of the maxim that correlation doesn't imply causation: differential privacy ensuring a lack of (strong) causation does not imply a lack of (strong) association. Our characterization also opens up the possibility of applying results from statistics, experimental design, and science about causation while studying differential privacy.

SoK: General Purpose Compilers for Secure Multi-Party Computation	Hastings et al., 2019	2019 IEEE Symposium on Security and Privacy	MPC	Secure multi-party computation (MPC) allows a group of mutually distrustful parties to compute a joint function on their inputs without revealing any information beyond the result of the computation. This type of computation is extremely powerful and has wide-ranging applications in academia, industry, and government. Protocols for secure computation have existed for decades, but only recently have general-purpose compilers for executing MPC on arbitrary functions been developed. These projects rapidly improved the state of the art, and began to make MPC accessible to non-expert users. However, the field is changing so rapidly that it is difficult even for experts to keep track of the varied capabilities of modern frameworks. In this work, we survey general-purpose compilers for secure multi-party computation. These tools provide high-level abstractions to describe arbitrary functions and execute secure computation protocols. We consider eleven systems: EMP-toolkit, Obliv-C, ObliVM, TinyGarble, SCALE-MAMBA (formerly SPDZ), Wysteria, Sharemind, PICCO, ABY, Frigate and CBMC-GC. We evaluate these systems on a range of criteria, including language expressibility, capabilities of the cryptographic back-end, and accessibility to developers. We advocate for improved documentation of MPC frameworks, standardization within the community, and make recommendations for future directions in compiler development. Installing and running these systems can be challenging, and for each system, we also provide a complete virtual environment (Docker container) with all the necessary dependencies to run the compiler and our example programs.

SoK: Understanding the Prevailing Security Vulnerabilities in TrustZone-assisted TEE Systems	Cerdeira et al., 2020	2020 IEEE Symposium on Security and Privacy	TEE	Hundreds of millions of mobile devices worldwide rely on Trusted Execution Environments (TEEs) built with Arm TrustZone for the protection of security-critical applications (e.g., DRM) and operating system (OS) components (e.g., Android keystore). TEEs are often assumed to be highly secure; however, over the past years, TEEs have been successfully attacked multiple times, with highly damaging impact across various platforms. Unfortunately, these attacks have been possible by the presence of security flaws in TEE systems. In this paper, we aim to understand which types of vulnerabilities and limitations affect existing TrustZone-assisted TEE systems, what are the main challenges to build them correctly, and what contributions can be borrowed from the research community to overcome them. To this end, we present a security analysis of popular TrustZone-assisted TEE systems (targeting Cortex-A processors) developed by Qualcomm, Trustonic, Huawei, Nvidia, and Linaro. By studying publicly documented exploits and vulnerabilities as well as by reverse engineering the TEE firmware, we identified several critical vulnerabilities across existing systems which makes it legitimate to raise reasonable concerns about the security of commercial TEE implementations.
TEE-Based Mutual Proofs of Transmission Services in Decentralized Systems	Liu et al., 2020	2020 IEEE Conference on Computer Communications Workshops	TEE	We propose a scalable and verifiable transmission recording system based on trusted execution environment (TEE) to support payment system for decentralized services. In the proposed system, consensus of the service is reached based on the service recording chains via mutual recording among participating nodes of the service; A simplified Merkle tree structure is used in the service records for checking the integrity of the transmission content, which facilitates efficient re-transmission of lost packets among neighboring nodes, and tracking of multi-path transmissions. The proposed system enables efficient and trusted incentive mechanisms to support network transmission services via edge devices (mobile devices, moving connected vehicles, base stations, etc.) in dynamic and self-organizing networks.
TOPPool: Time-aware Optimized Privacy-Preserving Ridesharing	Pagnin et al., 2019	2019 Proceedings on Privacy Enhancing Technologies	Homomorphic Encryption	Ridesharing is revolutionizing the transportation industry in many countries. Yet, the state of the art is based on heavily centralized services and platforms, where the service providers have full possession of the users' location data. Recently, researchers have started addressing the challenge of enabling privacy-preserving ridesharing. The initial proposals, however, have shortcomings, as some rely on a central party, some incur high performance penalties, and most do not consider time preferences for ridesharing. TOPPool encompasses ridesharing based on the proximity of end-points of a ride as well as partial itinerary overlaps. To achieve the latter, we propose a simple yet powerful reduction to a private set intersection on trips represented as sets of consecutive road segments. We show that TOPPool includes time preferences while preserving privacy and without relying on a third party. We evaluate our approach on real-world data from the New York's Taxi & Limousine Commission. Our experiments demonstrate

				that TOPPool is superior in performance over the prior work: our intersection-based itinerary matching runs in less than 0.3 seconds for reasonable trip length, in contrast, on the same set of trips prior work takes up to 10 hours.
Traffic Monitoring in Self-Organizing VANETs: A Privacy-Preserving Mechanism for Speed Collection and Analysis	Zhu et al., 2019	2019 IEEE Wireless Communications	Homo-morphic Encryption	With the explosive growth of vehicles, traffic monitoring has garnered significant attention in recent years. Collecting vehicular speed is an effective way to monitor traffic conditions and help vehicles to find optimal routes. However, further progress may be impeded due to users' privacy concerns. In addition, traffic monitoring is more difficult in a self-organizing VANET, since there is no centralized entity to collect and analyze the speed information. In this article, we mainly focus on privacy-preserving traffic monitoring in self-organizing VANETs. To address the unique features and security requirements of VANETs, we incorporate the homomorphic encryption, data perturbation, and super-increasing sequence in the proposed novel solution to resolve the challenges of efficient and privacy-preserving traffic monitoring. Security analysis shows that not only can our solution preserve vehicles' identities, locations, and data privacy, but it is also effective in mitigating collusion attacks. Moreover, experimental results confirm the efficiency of our solution in terms of computation and communication costs. Last but not least, some interesting challenges along with potential solutions are discussed, aiming to attract more research in this emerging area.
User Privacy Protection Method Based on Dynamic Hiding	Lou & Chen, 2018	2018 IEEE International Conference on Cloud Computing and Internet of Things	K-anonymity	The most commonly used method of location privacy protection is location K-anonymization. At present, most of the K-anonymization models are aimed at the attackers who do not understand the users' background knowledge. The probability of hacking will increase when they know about users. This paper proposes a multi-level meshing method to predict the user's trajectory according to the user's historical track data recorded by the LBS (location based service) server. Then the LBS server determines whether to dynamically adjust the location of the corresponding user in the K-anonymization model K-degree anonymous and remove the redundancy of the anonymous area while satisfying the K-degree anonymous. Due to the increase of the anonymous area, the impact on the quality of the LBS server is reduced. The experiments verify that this method is effective to the privacy protection of users when attackers know about the background knowledge.
VTDP: Privately Sanitizing Fine-grained Vehicle Trajectory Data	Liu et al., 2015	2015 Journal of LATEX Class Files	Differential Privacy	With the rapidly growing deployment of intelligent transportation systems (ITS) and smart traffic applications, vehicle trajectory data are ubiquitously generated, e.g., from GPS navigation systems, mobile applications, and urban traffic cameras. Analyzing such fine-grained data would greatly benefit the development of ITS and smart cities, yet pose severe privacy risks due to the recorded drivers' visited locations, routes, and driving habits. Recently, some privacy enhancing techniques were proposed to sanitize such data. However, such schemes have some major limitations – they either lack formal privacy

with Boosted Utility				notions to quantify and bound the privacy risks, or result in very limited utility, e.g., only a sequence of locations or aggregated information can be released (without retaining the speeds, accelerations and the timestamps of vehicles). In this paper, we propose a novel framework to sanitize the fine-grained vehicle trajectories with differential privacy (VTDP), which provides rigorous privacy protection against adversaries who possess arbitrary background knowledge. Our VTDP technique involves three phases of differentially private sampling, which sequentially generate all the three categories of data (besides a pseudo identity for each vehicle) – position, moving, timestamps. It also includes a vehicle trajectory interpolation procedure to further improve the output utility with the properties of fine-grained vehicle trajectory data. We conducted experiments on real vehicle trajectory datasets to validate the performance of our approach.
Mitigator: Privacy policy compliance using trusted hardware	Mazmudar & Goldberg, 2020	2020 Proceedings on Privacy Enhancing Technologies	TEE	Through recent years, much research has been conducted into processing privacy policies and presenting them in ways that are easy for users to understand. However, understanding privacy policies has little utility if the website's data processing code does not match the privacy policy. Although systems have been pro- posed to achieve compliance of internal software to access control policies, they assume a large trusted computing base and are not designed to provide a proof of compliance to an end user. We design Mitigator, a system to enforce compliance of a website's source code with a privacy policy model that addresses these two drawbacks of previous work. We use trusted hardware platforms to provide a guarantee to an end user that their data is only handled by code that is compliant with the privacy policy. Such an end user only needs to trust a small module in the hardware of the remote back-end machine and related libraries but not the entire OS. We also provide a proof-of-concept implementation of Mitigator and evaluate it for its latency. We conclude that it incurs only a small overhead with respect to an un- modified system that does not provide a guarantee of privacy policy compliance to the end user.

SGX-MR: Regulating Dataflows for Protecting Access Patterns of Data-Intensive SGX Applications	Alam, Sharma and Chen, 2021	2021 Proceedings on Privacy Enhancing Technologies	TEE	Intel SGX has been a popular trusted execution environment (TEE) for protecting the integrity and confidentiality of applications running on untrusted platforms such as cloud. However, the access patterns of SGX-based programs can still be observed by adversaries, which may leak important information for successful attacks. Researchers have been experimenting with Oblivious RAM (ORAM) to address the privacy of access patterns. ORAM is a powerful low-level primitive that provides application-agnostic protection for any I/O operations, however, at a high cost. We find that some application-specific access patterns, such as sequential block I/O, do not provide additional information to adversaries. Others, such as sorting, can be replaced with specific oblivious algorithms that are more efficient than ORAM. The challenge is that developers may need to look into all the details of application- specific access patterns to design suitable solutions, which is time-consuming and error-prone. In this paper, we present the lightweight SGX based MapRe- duce (SGX-MR) approach that regulates the dataflow of data-intensive SGX applications for easier application- level access-pattern analysis and protection. It uses the MapReduce framework to cover a large class of data- intensive applications, and the entire framework can be implemented with a small memory footprint. With this framework, we have examined the stages of data processing, identified the access patterns that need protection, and designed corresponding efficient protection methods. Our experiments show that SGX-MR based applications are much more efficient than the ORAM- based implementations.
Differentially Private Oblivious RAM Abstract:	Wagh, Cuff & Mittal, 2018	2018 Proceedings on Privacy Enhancing Technologies	TEE	In this work, we investigate if statistical privacy can enhance the performance of ORAM mechanisms while providing rigorous privacy guarantees. We propose a formal and rigorous framework for developing ORAM protocols with statistical security viz., a differentially private ORAM (DP-ORAM). We present Root ORAM, a family of DP-ORAMs that provide a tunable, multi-dimensional trade-off between the desired band- width overhead, local storage and system security. We theoretically analyze Root ORAM to quantify both its security and performance. We experimentally demonstrate the benefits of Root ORAM and find that (1) Root ORAM can reduce local storage overhead by about 2× for a reasonable values of privacy bud- get, significantly enhancing performance in memory limited platforms such as trusted execution environments, and (2) Root ORAM allows tunable trade-offs between bandwidth, storage, and privacy, reducing bandwidth overheads by up to 2×-10× (at the cost of increased storage/statistical privacy), enabling significant reductions in ORAM access latencies for cloud environments. We also analyze the privacy guarantees of DP-ORAMs through the lens of information theoretic metrics of Shannon entropy and Min-entropy [16]. Finally, Root ORAM is ideally suited for applications which have a similar access pattern, and we showcase its utility via the application of Private Information Retrieval.

Sajin Sasy* and Ian Goldberg* ConsenSGX: Scaling Anonymous Communications Networks with Trusted Execution Environments Abstract:	Sasy & Goldberg, 2019	2019 Proceedings on Privacy Enhancing Technologies	TEE	Anonymous communications networks enable individuals to maintain their privacy online. The most popular such network is Tor, with about two million daily users; however, Tor is reaching limits of its scalability. One of the main scalability bottlenecks of Tor and similar network designs originates from the requirement of distributing a global view of the servers in the network to all network clients. This requirement is in place to avoid epistemic attacks, in which adversaries who know which parts of the network certain clients do and do not know about can rule in or out those clients from being responsible for particular network traffic. In this work, we introduce a novel solution to this scalability problem by leveraging oblivious RAM constructions and trusted execution environments in order to enable clients to fetch only the parts of the network view they require, without the directory servers learning which parts are being fetched. We compare the performance of our design with the current Tor mechanism and other related works to show one to two orders of magnitude better performance from an end-to-end perspective. We analyse the requirements to actually deploy such a scheme today and conclude that it would only require a small fraction (<2.5%) of the relays to have the required hardware support; moreover, these relays can perform their roles with minimal network bandwidth requirements.
StealthDB: a Scalable Encrypted Database with Full SQL Query Support	Vinayagamurthy., Gribov, & Gorbunov, 2019	2019 Proceedings on Privacy Enhancing Technologies	TEE	Encrypted database systems provide a great method for protecting sensitive data in untrusted infrastructures. These systems are built using either special- purpose cryptographic algorithms that support operations over encrypted data, or by leveraging trusted computing co-processors. Strong cryptographic algorithms (e.g., public-key encryptions, garbled circuits) usually result in high performance overheads, while weaker algorithms (e.g., order-preserving encryption) result in large leakage profiles. On the other hand, some encrypted database systems (e.g., Cipherbase, TrustedDB) lever- age non-standard trusted computing devices, and are designed to work around the architectural limitations of the specific devices used. In this work we build StealthDB – an encrypted database system from Intel SGX. Our system can run on any newer generation Intel CPU. StealthDB has a very small trusted computing base, scales to large transactional workloads, requires minor DBMS changes, and provides a relatively strong security guarantees at steady state and during query execution. Our prototype on top of Postgres supports the full TPC-C benchmark with a 30% decrease in the average throughput over an unmodified version of Postgres operating on a 2GB un- encrypted

Table 3. Summary of reviewed literature

4.2 De-Identification Techniques

In the following the main de-identification techniques used in this work are explained in more detail. For each technique, an example is provided on how the technique might be used in the mobility domain. The main results of the literature review are explained in more detail in order to establish how exactly the techniques are currently being researched in academia. This is done to demonstrate the advancements that are being done for each technique.

4.2.1 Differential Privacy:

Differential privacy can best be defined as a system to share information in a dataset while withholding information about a single information in that dataset. Algorithms are defined as being differentially private if one cannot tell if a data entry from a specific individual is included in a dataset or not. Thereby, differentially private algorithms resist de-identification or re-identification attacks. Differential privacy is not a de-identification technique itself but rather a measurement for privacy whereby techniques such as noise addition are used to create differential privacy. The concept was first described by Cynthia Dwork in 2006 [1], has been improved by other researches and implemented in applications such as the machine-learning libraries Pytorch and Tensorflow[2]. Differential privacy offers a clear benefit of increasing privacy while decreasing the accuracy of the output. Thus, its application lies more in the detection of trends in use cases where anomalies or strong outliers are less relevant.

Example:
A vehicle fleet owner would like to know the average driven kilometer from their 10 vehicles. However, the actual driven kilometer per vehicle and the corresponding employee needs to be kept private. Let us say that all vehicles have driven between 40,000 and 50,000km.

A trusted third party could know be used to calculate the average over all vehicles. However, should the fleet owner or one of the drivers already know of a majority of the exact driven kilometers, the exact driven distance of the remaining vehicle could also be deduced. Therefore, the third party adds noise to the data by exchanging one of the actual values with for instance a random value in the range of all values. This slightly changes the average over all vehicles but makes it impossible to derive the actual exact value of a single vehicle.

[2] For a short guide on differential privacy in Tensorflow see: Radebaugh & Erlingsson, 2019. Introducing TensorFlow Privacy: Learning with Differential Privacy for Training Data https://blog.tensorflow.org/2019/03/introducing-tensorflow-privacy-learning.html (Last visited, 25.02.2020)

Literature review results

The following provides exemplary findings of the literature review on differential privacy in information systems.

[2] provides an introduction to differential privacy in the automotive domain. The authors introduce differential privacy as a concept and relate it to mobility use cases. Additionally, recommendations are given on how best to tackle privacy in such use cases. The desired level of privacy should be stated and which and whether other parties can be trusted. A privacy budget needs to be introduced and data should be protected by enforcing differential privacy directly within the vehicle. As challenges, the authors identify the correct setting of a privacy budget as well as the multidimensional time series nature of the data that can only allow for event-level privacy. The work of [3] provides an analysis on differential privacy in general.

The authors in [3] propose a differential privacy protection method for frequent pattern mining in view of the application-level privacy protection requirements of industrial inter-connected systems. This method designs a low-cohesion algorithm to realize differential privacy protection. In the implementation of differential privacy protection, Top-k frequent mode method is introduced, which combines the factors of index mechanism and low cohesive weight of each mode, and the original support of each selected mode is disturbed by Laplacian noise. It achieves a balance between privacy protection and utility, guarantees the trust of all parties in cyber-physical systems and provides an effective solution to the problem of privacy protection in industrial internet systems. The proposed approach provides better performance in terms of false-negative rate and average relative error.

The work of [4] studied unlinkability based on differential privacy in Vehicle Ad-Hoc Networks (VANETS). Hereby, a vehicle may communicate with other vehicles or road infrastructure. The work demonstrates that by using pseudonym swaps to project vehicle trajectory privacy in such VANETs, existing solutions make it possible to link old and new pseudonyms of vehicles. To overcome this issue the authors, use differential privacy for pseudonym swaps. Using a real dataset, the authors find that their solution ensures unlinkability of pseudonyms. The solution is also found to require 50% less pseudonyms and the successful swap process of pseudonyms works with a probability of more than 90%, representing a higher success rate than other works.

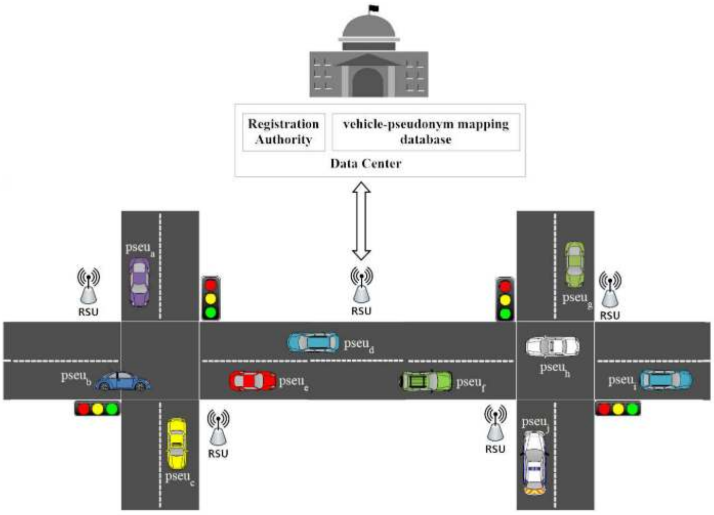


Figure 6. Pseudonym swap architecture by Li et al., 2020

[5] does also study differential privacy in VANETs. The authors add noise to the vehicle's side to achieve privacy. The proposed solution scales the noise with respect to the data correlation of the vehicles and can be adapted towards different types of data. Most importantly, the proposed solution does not rely on a trusted party but select a leader out of the vehicles that then transfers the data to the controller which then processes the data. Before transferring the data, the group leader vehicles perform multiple computations on the data: compression, perturbation and filtering.

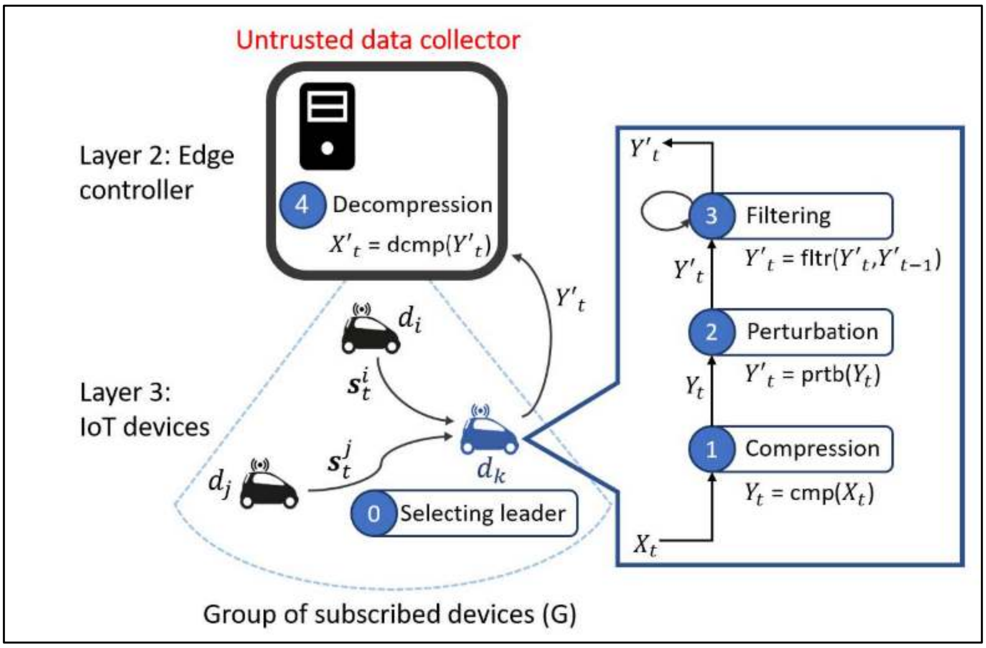


Figure 7. Differentially Private Data Streams design by [5]

The authors test the model by using a simulated dataset by the institute TAPASCologne that consists of mimicked traffic data (speed and location) from Cologne, Germany. The results demonstrate that considering data properties in "calibrating the noise significantly improves the utility when the noise is added in the IoT layer" [5].

[6] also investigated differential privacy for traffic volume estimation. The authors find that their scheme can protect location and trajectory privacy of vehicles, as verified by theoretical analyses and simulations.

The work of [7] provides an interesting modification compared to the works discussed above, in that it includes the assumption that adversaries may possess background knowledge. The authors implement differential privacy for vehicle location data, finding that their solution protects against re-identification and produces high output utility. The work of [8] provides a similar approach towards differential privacy in vehicles as the research discussed above.

4.2.2 Homomorphic Encryption:

Homomorphic encryption is a special form of encryption, the encoding of information, whereby it is possible to perform computations on the encrypted data without the need to decrypt it first. This allows for a high level of privacy as data might be transferred to third parties which can then process the data without gaining personal information and insights from it. Such third parties might be data analytics providers and cloud services that have greater storage or higher processing capabilities than oneself. Homomorphic encryption might also be useful in instances where the data controller or data steward does not have the permission to transfer personal data. The technique itself was originally introduced in 1980s next to the development of the "RSA" cryptography. The technique was continuously improved and in 2009 the first fully homomorphic encryption (FHE) was introduced by Craig Gentry [9]. It can be seen as an extension of public-key cryptography. However, to this day, the technology is still limited by the lack of standards as well as its efficiency and effectiveness in its applicability. Only a limited set of computations, usually additions or multiplications, are possible and computations on the encrypted data are much slower and computationally expensive. Thus, possible use cases need to be less time sensitive and to be performed on capable hardware.

Example:
A vehicle is creating a status report on its mechanical parts. This report is then homomorphically encrypted and send to a data analytics provider. This provider is only able to see the encrypted data from which no further information can be derived. The provider performs several, previously discussed, computations on the encrypted data, as if the data was not encrypted. The results are sent back to the vehicle that can now decrypt the data. The results are the same as if the operations had been performed on the decrypted data.

Literature review results
The following provides exemplary findings of the literature review on differential privacy in information systems.

The authors of [10] introduce homomorphic encryption in VANETs. The authors' solution creates self-generated randomized pseudo-identities for vehicles. However, a trusted authority is needed in their framework.

In [11] the authors investigate vehicle-based spatial crowdsourcing (SC) applications whereby workers have to upload their driving locations. They propose a decentralized, privacy-preserving solution for such Internet of Vehicles SC using homomorphic encryption, zero-knowledge proof and circle-based location verification. Using blockchain technology, a decentralized network is constructed by road side units. The model is shown in Figure 8.

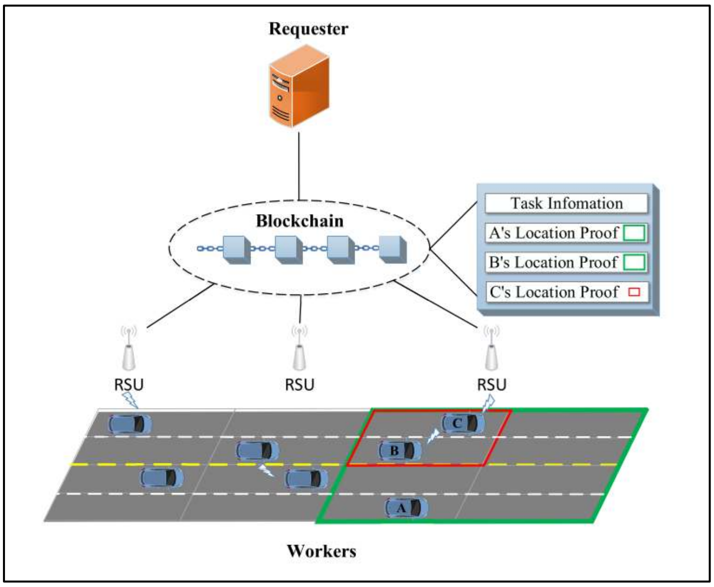


Figure 8. System model using blockchain technology by [11]

The solution results in a grid where workers are located in, whereby the size of the grid demonstrates the achieved level of privacy. The authors state that their solution is efficient and feasible in real-life use cases. In future work, the authors aim to study how to also protect task and solution privacy of workers, not just location information.

The authors of [12] also use homomorphic encryption for VANETs and find their solution to be efficient and secure. In their setup, both RSUs and location-based services providers are untrusted entities that may be compromised or track vehicles and drivers. Upon evaluation, the authors find that their solution has a complexity of search efficiency of 0(logN), lower than other homomorphic approaches by other researchers.

Similar work using homomorphic encryption has been done in [13] and [14] in which the authors of [14] combine homomorphic encryption and differential privacy in order to create an energy-efficient privacy scheme for VANETs.

4.2.3 K-anonymity:

Just like differential privacy, k-anonymity is not a de-identification technique in itself but rather a property that anonymized data might possess. First described in 1998, k-anonymity defines a state where a person cannot be distinguished from k-1 other persons in a dataset [15]. This dataset then has achieved k-anonymity. Several techniques such as data suppression and generalization are used to create a k-anonymous dataset. However, while k-anonymity is a comparatively simple concept that is easy to implement, there are multiple attacks that enable re-identification. In case of identical data records, k-anonymity would be achieved although a person could still be identified from the data (homogeneity attack). Additionally, external data and information can be used to re-identify a person in an otherwise k-anonymous dataset as well as background knowledge on relationships between sensitive attributes and quasi-identifiers contained in a dataset. Due to these shortcomings, several extensions such as l-diversity and t-closeness have been derived that add further limitations to a dataset.

Example:
A dataset contains the values seen in the following table:

Vehicle owner	Age	Vehicle parking location	Gender
Alice	25	Frankfurt	Female
Bob	45	Frankfurt	Male
Cindy	54	Berlin	Female
Dirk	42	Frankfurt	Male
Esther	40	Berlin	Female
Fiona	21	Göttingen	Female

Table 4. Plain dataset including identifiers and sensitive information

The dataset is not anonymized as each individual could easily be identified given their specific attributes. Using suppression and generalization we can at least achieve a weak form of k-anonymity.

Vehicle owner	Age	Vehicle parking location	Gender
-	18 - 39	Frankfurt	Female
-	40 - 60	Frankfurt	Male
-	40 - 60	Berlin	Female
-	40 - 60	Frankfurt	Male
-	40 - 60	Berlin	Female
-	18 - 39	Göttingen	Female

Table 5. Anonymized dataset that achieves k=2-anonymity

As can be seen, at least 2 records exist that are similar for the attributes age and gender, achieving 2-anonymity.

Literature review results
The following provides exemplary findings of the literature review on differential privacy in information systems.

In [16], the authors achieve privacy protection through k-anonymity in autonomous vehicles. In the paper, a new notion of client-based personalized k-anonymity (CPkA) is introduced. The others use two models to achieve utility and privacy in autonomous vehicles. Here, in-group mechanisms and optimal grouping strategies are combined. An autonomous driving vehicle is querying information such as a destination for a cyber-physical system (CPS) provider and needs to send that service provider its own information, such as the geolocation of the vehicle, in order to obtain this service. K-anonymity is newly introduced in that the vehicle creates dummy query content and sends multiple queries at once. The service provider now does not know which query is the valid one and has to respond to each query, while the vehicle can now use the correct information and disregard the other queries. The paper even assumes that there is a strong attacker that has prior knowledge on the anonymization mechanism. The authors test their CPkA mechanisms using real-life datasets, from OpenStreetMap among others, and demonstrate the effectiveness and efficiency of their mechanisms. However, it has to be noted that geolocation data is only divided into 30km x 30km squares, creating very large locations in which car data is protected using k-anonymity.

[17] provides a more general overview on k-anonymity in VANETs. The four methods hiding, obfuscation, anonymizing and dummifying can be used to achieve k-anonymity in vehicle location data. However, all methods lead to a decrease in quality of service and/or safety as the true information and its frequency are rendered less useful. The authors furthermore define macroscopic and microscopic location privacy. Microscopic location privacy is defined as anonymity at a specific time and location while macroscopic location privacy means anonymity from the beginning to the end of a path. The authors introduce a new model KTD, based on k-anonymity whereby K is the number of entities that might be confused with each other at a time t, the anonymity duration T and the average distance deviation D between entities. They tested the new model on five different privacy protocols in a 3km grid with up to 897 virtual vehicles to assess its performance. The five protocols are as follows:

"SMP-R, stationary mix points, occurring at regular time intervals; SMP-I, stationary mix points, occurring at irregular time intervals; OTFP-R, randomly chosen on-the-fly mix points, occurring at regular time intervals; OTFP-I, randomly chosen mix points, occurring at irregular time intervals; and GLRP-2, group leader relay points, which occur continuously throughout the trajectory of a vehicle designated as the leader of a group of vehicles traveling within range of the leader. The number 2 in GLRP-2 indicates that vehicles join the group in pairs" [17]. It can be seen that the average K increased with the density of vehicles in a region while the average distance deviation in an anonymity set of vehicles varied depending on the density. Here, some protocols performed better than others, with a group leader protocol performing noticeably worse than other protocols.

4.2.4 Secure Multiparty Computation:

Secure Multiparty Computation (SMC) allows for the sharing of information without the need for a trusted third-party while maintaining privacy at the same time. SMC is used to analyze data from multiple different parties whereby every party is able to keep its information secret.
The technology is based on secret sharing, sensitive data from each entity is encrypted and distributed to other entities. One such secret in its own is worthless as no information can be derived from it; only in combination with the other shares, across multiple parties, can it be used. The concept of SMC has existed for decades and is continuously being improved by researchers to increase efficiency and scalability. Its applicability is best in use cases in which data needs to be computed and aggregated over multiple untrusted entities or the lack of a trusted third-party. A drawback of SMC are high set-up costs as the solution needs to be specifically customized to the use case and cannot be adapted as easily as for instance the model

of differential privacy. Additionally, computations are time- and resource-intensive as they are distributed over multiple entities.

Example:

Consider the three vehicles A, B and C below. Each vehicle is reporting a specific value, e.g., the distance driven in the last hour (5km, 10km and 15km). The vehicles would like to know the average distance over all vehicles without revealing their own distance and without a third-party (e.g. (5 + 10 + 15)/3 = 10). SMPC is used to solve this problem.

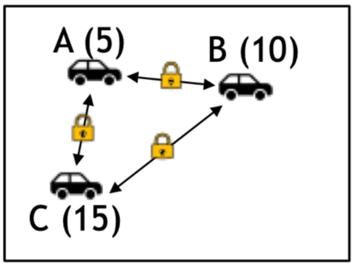


Figure 9. First step of an example of SMC using three vehicles. Each vehicle reports an individual value

Each vehicle randomly selects two numbers between 0 and 3, representing the upper limit of participants in this example, with which the true value of the specific vehicle is then multiplied. These values are distributed to the other vehicles (e.g., A distributed 6.5 to B and 9 to C). Now, each vehicle takes their true value and adds the values given to it by the other vehicles. Then, the values that the vehicle itself distributed to the others are subtracted. This generates a new value per vehicle that again can be shared with the other vehicles. The average over these numbers is the same as the average over the initial true values of all vehicles, as can be seen in the calculations below.

	A	B	C
A provides (1.3/1.8)		$5 * 1.3 = 6.5$	$5 * 1.8 = 9$
B provides (0.4/1.2)	$10 * 0.4 = 4$		$10 * 1.2 = 12$
C provides (0.5/1.5)	$0.5 * 15 = 7.5$	$1.5 * 15 = 22.5$	

A reports: $5 + 4 + 7.5 - 6.5 - 9 = 1$
B reports: $10 + 6.5 + 22.5 - 4 - 12 = 23$
C reports: $15 + 9 + 12 - 7.5 - 22.5 = 6$

$$\frac{Report\ A + Report\ B + Report\ C}{|vehicles|} \Leftrightarrow \frac{1 + 23 + 6}{3} = 10$$

Table 6. Calculations used for SMC

Literature review results

The following provides exemplary findings of the literature review on differential privacy in information systems.

Secure Multiparty Computation (MPC) allows for computation of several functions, directly on encrypted data, guaranteeing that the data is kept private from the parties doing the computations. Hereby, an entity divides the secret, the private value(s) into a given number of fragments, shares, which are then distributed between several entities. These computing parties perform MPC collaboratively, whereby the parties, or nodes, have to exchange information between multiple MPC rounds among each other. The secrets and the results of the computation are then distributed to one party that can then reveal the encrypted result of the computation. Figure 10 by [18] depicts a generalized MPC architecture.

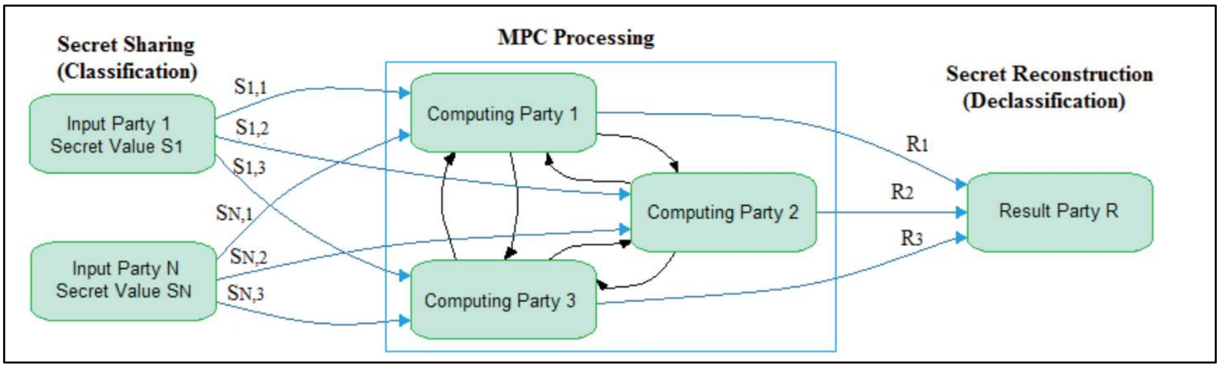


Figure 10. General MPC architecture using secret sharing by [18]

The authors of [18] evaluate and compare two MPC sub-protocols in order to identify whether communication cost or the number of calculation rounds have a higher impact on the performance of the subprotocols. Communication rounds are the number of times computing nodes send a message to each other while communication costs are the number of values exchanged between the computing nodes during an operation. They find that performance is more strongly influenced by communication costs than by the number of calculation rounds. This is because “by trying to minimize the number of rounds, the complexity of the algorithms increases, messages that contain more values are exchange[d], and a larger quantity of data should be processes by the computing nodes” [18].

In [19], the authors investigate MPC for IoT use cases and create a solution that allows third parties to query data and include measures for access control and inform data sources about query requests and the usage of their data. This solution offers sensors or devices the opportunity to intervene and reject processing requests if such requests do not fulfill the desired privacy requirements of the sensor.

The work of [20] provides a much-needed overview on general purpose compilers for MPC. In academia, several general-purpose MPC solutions have been developed in the past. However, these solutions are generally not highly efficient due to computation and communication complexity. In order to increase efficiency, solutions have been built for very specific use-cases. While these custom-made solutions are more efficient, they are not scalable and adaptable towards other use cases. The authors argue that general-purpose compilers can help. The authors identify and evaluate eleven such compiler systems and find that is challenging to run and use the systems. As an additional contribution the authors provide virtual environments for the systems for future readers and users.

In [21], the authors combine MPC and homomorphic encryption and evaluate their privacy-preserving solution. The authors find that the solution takes 0.2 seconds for multiparty key generation while the multiparty decryption takes 0.06 seconds. The required memory for encoding depends on “the security parameter and the maximum number of levels of the evaluation circuit, where the size of encoding of a secret key is in the range [16KB-13MB]" [21]. Computation times for key generation and decryption depend on the number of parties and encoding size, while the communication cost is linear in N.

In [22], the authors study deep learning and use MPC to solve privacy issues in deep learning training data. The authors use benchmark datasets like MNIST to demonstrate that MPC can secure training data. However, it is noted that this privacy-friendly solution is very time consuming and costly.

4.2.5 Federated learning:

In comparison to distributed machine learning that is widely used today to solve performance problems [23], Federated Learning (FL) also has clear focus on privacy protection. Originally proposed by Google in 2016 [24], the idea of federated learning aims to build an machine learning model based on datasets distributed across multiple devices, where the data is not merged into one overall dataset. Meanwhile, there exist different approaches and typologies such as horizontal FL, vertical FL and federated transfer learning, like in Yang et al. [25].

Thereby the horizontal FL architecture with a central server that controls the main model and a local model on each end user device is most common (see Figure 11). The local model is a copy of the main model on the central server that is updated from time to time.

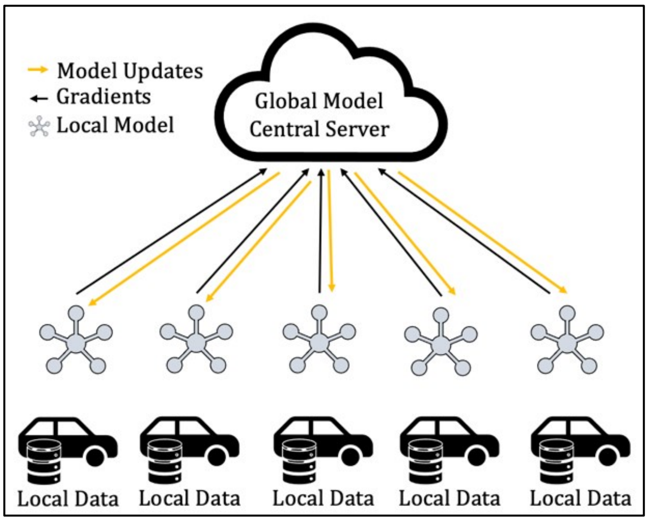


Figure 11: Horizontal Federated Learning Architecture (compare [26]*)*

Each local device collects data and improves the local model by local model updates. To update the model on the central server, the gradients of the local model can be sent to the central server periodically, based on batches, thresholds or central server request (see Figure 12). In horizontal FL all entities share the same feature space.

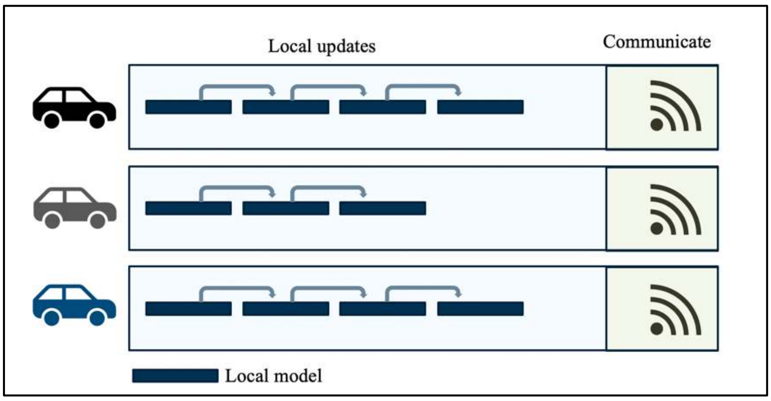


Figure 12: Parameter updating in a federated learning architecture (compare [26]*)*

Since only necessary updates are transmitted between the local model and the main model, this approach has two advantages. First, there is no central database, which makes it immensely difficult for potential attackers to obtain large amounts of data. Second, only fragments of the locally stored data are transferred to the main model [26].

Figure 13 provides an overview of the vertical FL that is mostly used to train a joint model among companies that own different feature values for certain instances that they want to combine to train a joint model. Therefore, the feature space is different.

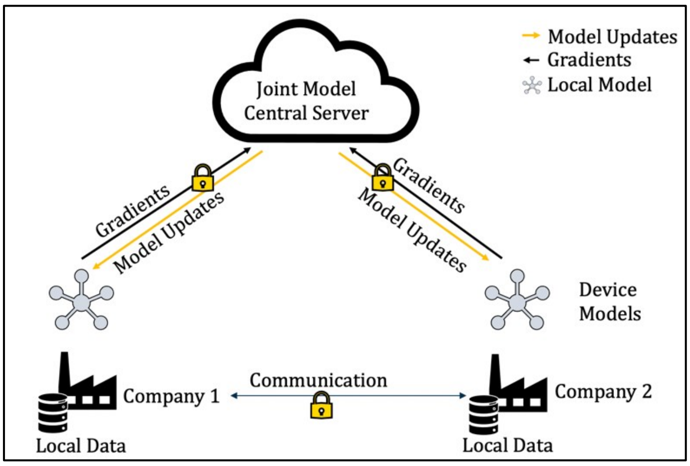


Figure 13: Vertical Federated Learning Architecture (compare [26])

This method is less privacy preserving because the data is already collected and stored in a company. FL is mostly used to train models among industries that are not allowed to share data with each other, e.g., finance and retail.

Literature review results

Liu et al. [27] propose a traffic flow forecasting approach based on federated learning that is based on large volume data gathered by organizations and government and contains a lot of private user data. In their study, they introduce an FL-based gated recurrent unit neural network algorithm that exhibits a high accuracy while at the same time communication overhead is reduced and the data privacy is not compromised.

A similar approach is proposed by Xu and Mao [28] who introduce a software-based traffic congestion monitoring system. In their study, federated learning is especially used to identify vehicle targets in remote sensing images.

Khan et al. [29] propose a federated learning model to satisfy the demand of a strong interplay between the key stakeholders, such as city authorities and communication network providers in an autonomous driving environment. Their model supports achieving traffic efficiency and the resource allocation of the network provider.

One of the most relevant fields in the future is in the field of medicine. While considerable success has already been achieved in deep neural network (DNN) training, AI requires larger amounts of imaging and clinical data for reliable clinical decision support. This data cannot be obtained in voluntary clinical trials at a small number of institutions that are not well distributed geographically. This problem is exacerbated by regulations such as the GDPR or the United States Health Insurance Portability and Accountability Act (HIPAA), which strictly regulate the sharing and storage of personal data [30], [31], [32], [33]. This is where federated learning comes into play, ensuring data privacy on the one hand and data usage between institutions on the other.

Another market with great potential is the financial sector. A large number of factors are combined in the prediction of credit risks. Although efficient intra-bank ML systems exist, a huge efficiency gain can be expected for inter-bank models. Federated learning can allow banks to share information about their customers' credit risk while keeping privacy-sensitive data local and invisible to other banks [34], [35].

In [2], the authors study the problem of federated learning training over a flat-fading Gaussian multiple access channel (MAC), subject to local differential privacy constraints. They propose and study analog aggregation schemes, in which each user transmits a linear combination of a) local gradients and b) artificial Gaussian noise, subject to power constraints. The local gradients are processed as a function of the channel gains to align the resulting gradients at the parameter server, whereas the artificial noise parameters are selected to satisfy the privacy constraints. The proposed approach decreases the training error as the number of users increases and converges to the centralized algorithm in which all points are available at the parameter server.

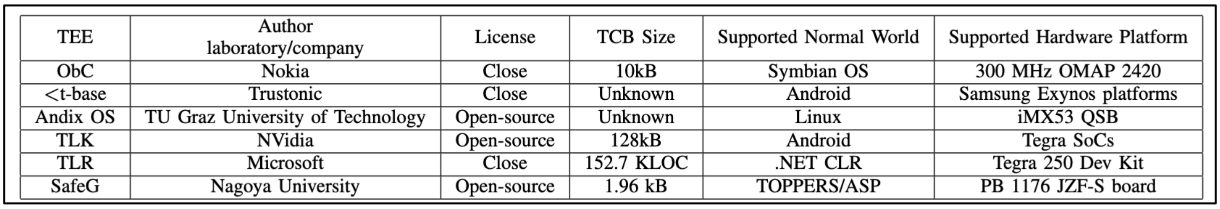
4.2.6 Trusted execution environment:

A trusted execution environment (TEE) is a dedicated secure area and execution environment within an untrusted piece of hardware. As multiple definitions on TEE exist, we follow the general definition of [36] that defines TEE as “an execution environment which protects both its runtime states and stored assets, hence the need for isolation and secure storage”. The TEE runs on a separation kernel, guaranteeing the authenticity of the executed code, protecting its execution against software and physical attacks performed from outside the TEE.

Literature review results

The following provides exemplary findings of the literature review on differential privacy in information systems.

[36] provides a first introduction on TEE and aims to create a common definition for it. The authors furthermore compare and analyze several existing models for the first time. Table 7 depicts the different TEE that have been compared.



TEE	Author laboratory/company	License	TCB Size	Supported Normal World	Supported Hardware Platform
ObC	Nokia	Close	10kB	Symbian OS	300 MHz OMAP 2420
<t-base	Trustonic	Close	Unknown	Android	Samsung Exynos platforms
Andix OS	TU Graz University of Technology	Open-source	Unknown	Linux	iMX53 QSB
TLK	NVidia	Open-source	128kB	Android	Tegra SoCs
TLR	Microsoft	Close	152.7 KLOC	.NET CLR	Tegra 250 Dev Kit
SafeG	Nagoya University	Open-source	1.96 kB	TOPPERS/ASP	PB 1176 JZF-S board

Table 7. Different types of TEEs compared by [36]

The work of [37] proposes a transmission recording system based on TEE for payment systems for decentralized services. The proposed framework therefore uses TEEs to create verifiable service records chains, using a Merkle tree structure and a blockchain-based payment system. An overview of the system is shown in Figure 14. The solution is found to be scalable and offers consensus through the blockchain solution as multiple nodes validate services. Service records and updates are calculated in the TEE.

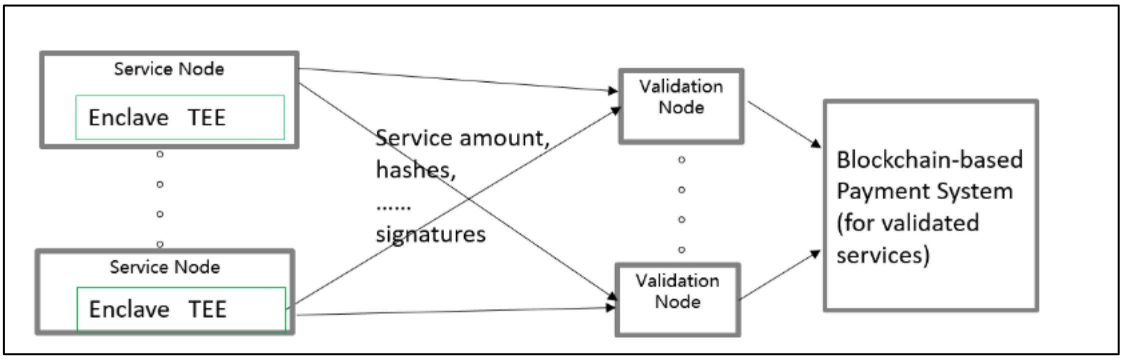


Figure 14. System overview for TEE-based transmission service, by [37].

The authors of [38] propose StealthDB, an encrypted cloud database build using Intel SGX TEE. The database offers full SQL support and can run on any newer Intel CPU. The system is scalable and comes with a 30% decrease in throughput, representing about 1ms latency increase. The source code of the prototype is provided as open source. In Figure 15 and Figure 16 the high-level and actual StealthDB architecture are shown.

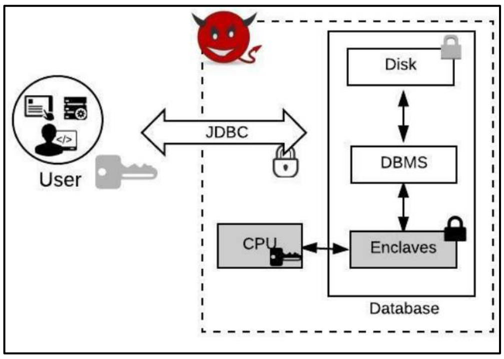


Figure 15. StealthDB High-Level architecture by [38]

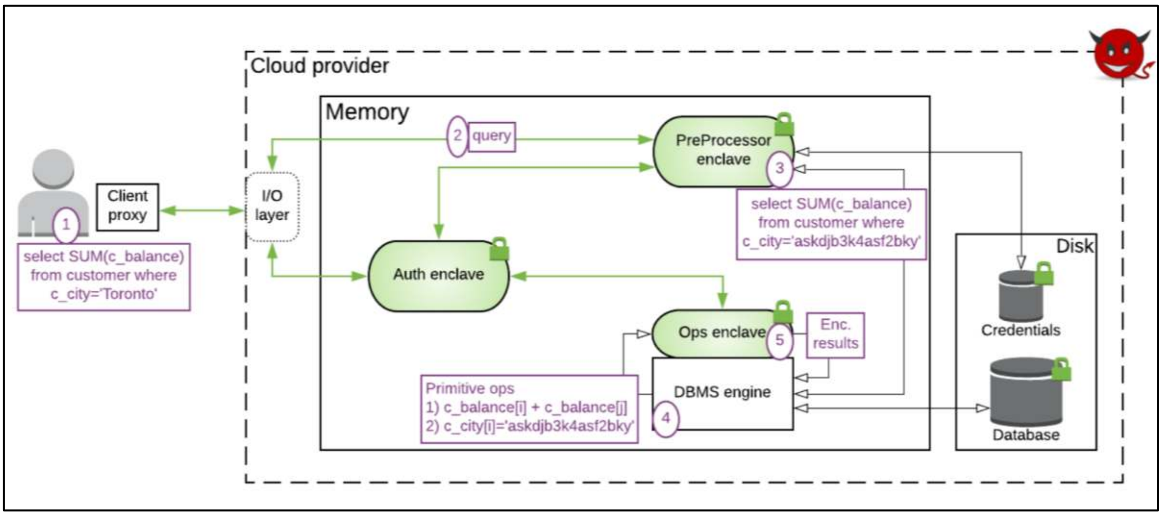

Cloud provider
Memory
2
query
PreProcessor enclave
3
Client proxy
I/O layer
select SUM(c_balance) from customer where c_city='askdjb3k4asf2bky'
1
select SUM(c_balance) from customer where c_city='Toronto'
Auth enclave
Disk
Credentials
Ops enclave
5
Enc. results
Primitive ops
1) c_balance[i] + c_balance[j]
2) c_city[i]='askdjb3k4asf2bky'
DBMS engine
4
Database

Figure 16. StealthDB architecture by [38]

The work of [39] defines differentially private Oblivious RAM, or ORAM, protocols. The created protocols are evaluated in a TEE and a server-client setting. ORAM is a cryptographic primitive that allows to protect data access patterns on an untrusted server. One use case of ORAM is the possible mitigation of side-channel attacks in TEE. However, the authors acknowledge that their solution is not suitable for all use cases, requiring significant overhead, and may be seen as a first step for further research. The research of [40] goes in a similar direction by looking at how best to protect privacy of access patterns in Intel SGX TEE. The authors agree that ORAM provides a very expensive solution and propose a different framework, called SGX-MR (MapReduce) that works for many data-intensive applications. In comparison to ORAM, the authors' solution is found to be much more efficient.

In [41], the authors focus on TEEs build with Arm TrustZone and evaluate security issues of such systems by large manufacturers such as Qualcomm, Trustonic, Huawei and Nvidia. In their work, the authors identify several security issues of such TEEs for commercial implementation. As a conclusion, the authors argue that the common belief that TEEs are secure is questionable due to the number and diversity of found security issues in different TEEs. Lastly, the paper provides several recommendations on how to counteract such security issues.

4.2.7 Overall conclusion

By evaluating current academic literature, we could see that progress is being made with respect to de-identification methods. Apart from the techniques identified in ISO/IEC 20889:2018. Most notably, TEEs, federated learning and MPC are techniques that might be beneficial for different use cases in the mobility domain. Homomorphic encryption and differential privacy are already included in ISO/IEC 20889:2018 and are continuously being researched and improved upon as both techniques offer promising privacy guarantees. Furthermore, academic research demonstrated a strong focus on VANETs, likely due to the interesting scenario of privacy protection given a continuous change in locations and the number of entities, vehicles, in a use case. This area of research serves as a strong basis of knowledge for the use cases in the upcoming chapters that focus on information exchange between multiple vehicles.

Most notably, it could be seen that scientists often aim to combine several techniques and their respective strengths, or to counteract weaknesses and open issues that a specific technique may have. Naturally, this increases the complexity of a solution, making its evaluation more challenging.

5 WP3

The aim of this working package was to evaluate the de-identification methods identified in WP2 upon the use cases initially developed in WP1. Thus, this section includes the technical evaluation of suitable de-identification techniques for use cases with respect to predictive maintenance, autonomous driving and data sharing and analysis with third parties.

To do so, external databases have been identified as a first step. One possible attack vector that might arise when working with data is the re-identification of data using such external databases as they might contain information with which data in the developed use cases might be combined.

Upon identifying these databases, each use case will be analyzed in detail. Here, a scenario description, data flow charts and assumptions and requirements for each use cases are described in detail. Using exemplary attack scenarios, we identify suitable de-identification techniques that are able to provide an acceptable level of data privacy. For each technique, the level of effort and quality of results are determined. Hereby we focus on a qualitative evaluation based on academic literature without the use of real data. All de-identification techniques and the proposed solutions based on them are evaluated on the following aspects:

- Protective effect: *The overall level of privacy that can be achieved through the de-identification technique in the particular use case. An optimal solution is able to protect personal information against any attack scenario outlined in this work.*
- Complexity: *Complexity describes the overall complexity to develop, implement and maintain a particular solution. Oftentimes, a de-identification technique cannot simply be put to work but requires careful finetuning towards the specific type and frequency*

of gathered data as well as the desired output. Additionally, techniques and their algorithms need to interact with the environment in which they are implemented.

- Runtime: *Runtime describes the time that the overall solution for a use case needs to perform all necessary tasks that lead to the de-identification of data. This includes the actual runtime of algorithms, the execution of code, and the gathering and distributing of data and results between different entities.*
- Degree of maturity: *The degree of maturity describes the scientific and commercial advancement of a de-identification technique. While some techniques are already used regularly, others need not yet be suitable for commercial use.*
- Implementation effort: *The overall effort that needs to be taken to implement the solution for a specific use case. This includes the provision, installation and finetuning of hardware and software for the specific entities as well as the time and human resources that are needed for its implementation.*
- Monetary cost: *Monetary cost includes the cost of development and procurement of all necessary hard- and software for each use case.*
- Possible interfering factors: *Interfering factors are use case-specific circumstances and factors that might hinder the performance, effectiveness and efficiency of the de-identification technique.*

The quality of the data after the implementation of the de-identification techniques are evaluated on the following aspects:

- Time blur: *Time blur depicts the degree to which data loses information that are related to a specific time point. That means data that is gathered over a period of time and might then be aggregated to a single data point. Here, time-related information gets lost, resulting in time blur.*
- Time delay: *Time delay depicts the delay with which data is reported and can be acted upon. That is, data might be collected continuously but loses its value as the computation of results takes significant time, resulting in a time delay that decreases the value of created insights.*
- Location blur: *Location blur depicts the degree to which location data gets blurred in a specific scenario. Location data might for instance be aggregated on a street, city or kilometer-basis.*
- Processing speed: *Processing speed describes the execution time of the de-identification technique itself.*
- Aggregated data: *Aggregated data describes a state in which data that is gathered during a use case is aggregated and thus a loss of information in the data occurs. While most scenarios allow for some aggregation, as the amount of data that is produced is high, more aggregation is likely to decrease the usability of a de-identification technique.*
- Truthfulness: *Truthfulness describes whether input data and output data are equal when using a de-identification technique. Different techniques may report non-truthful data when data is perturbed, noise is added or the sequence of data is changed. Less truthful data output can decrease validity of insights that are generated in a use case.*

The combined evaluation of the different aspects described above enables us to make a statement on the overall suitability, and usability, of a de-identification technique for a particular use case. For each use case, a table is provided that compares all suitable de-identification techniques against each other. Factors are ranked as *Low*, *Medium* and *High* whereby a color-code using red, yellow and green demonstrates the positive or negative effect of that ranking. For instance, a technique may score *High* on complexity, which would result in a red color-code as a high degree of complexity is not seen as favorable.

5.1 External Databases

Even when data has been, supposedly successfully, anonymized and personal data cannot be gathered from a database, the threat of re-identification through the combination of the data using external databases still remains. It could be seen that re-identification is possible using publicly available information [3]. This demonstrates that external databases need to be considered when working with personal data. Such databases may be publicly available online, or belong to organizations that offer access to data as a service.

Under consideration of the use cases that are to be evaluated in this report, a review has been conducted to gather information on databases, and types of data, that may pose a re-identification threat.

Table 8 consists of a shortlist of different existing databases that contain location and mobility related data. For the use case of the electricity grid provider, one example of weather data is also included. Additionally, an exemplarily list of simulated traffic flow databases is provided. Such databases provide the opportunity to configure different traffic scenarios. For instance, the influence of different geographic and topographic factors on traffic flow might be analyzed.

Database	Provider	Link	Country	Data
Digital Recognition Network	DRN	https://drndata.com/	US	location, license plate, vehicle owner data, street data
Zentrales Fahrzeugregister	Kraftfahrtbundesamt	https://www.kba.de/DE/ZentraleRegister/ZFZR/zfzr_node.html	DE	vehicle data, vehicle owner data, license plate, insurance

[3] Simon et al., 2020. Toolkit for assessing and mitigating Risk of re-identification when sharing data derived from health records. Available under: https://www.sentinelinitiative.org/sites/default/files/Methods/Sentinel_Report_Toolkit-Assessing-Mitigating-Risk-Re-Identification-Sharing-Data-Derived-from-Health-Records.pdf (Last accessed: 14.05.2021)

				information, audit reports
onlinestreet	Hello World Digital	https://onlinestreet.de/	DE	street map
Das Telefonbuch	Deutsche Tele Medien GmbH	https://www.dastelefonbuch.de/	DE	Addresses, name
Traffic Stats	TomTom	https://www.tomtom.com/products/historical-traffic-stats/	NL	traffic information
Google Maps	Google	https://www.google.de/ https://www.google.de/maps	US	street map, traffic flow information
Apple Maps	Apple	https://www.apple.com/de/maps/	US	street map, traffic flow information
AutoDNA	ASDIRECT	https://www.autodna.com	PL	VIN information
Here	Here	https://www.here.com/solutions/automotive	US	street map
Meteostat	Meteostat	https://meteostat.net/de	DE	weather data
OpenStreetMap	OpenStreetMap Foundation	https://www.openstreetmap.org/#map=6/51.330/10.453	UK	street map
Vinencoder	Vincario s.r.o.	https://vindecoder.eu	CZE	VIN information
Traffic Simulation				
TapasCologne Project	German Aerospace Center (DLR) and others	https://sumo.dlr.de/docs/Data/Scenarios/TAPASCologne.html	DE	street data, simulated traffic flow
TSIS CORSIM	Mc Trans	https://mctrans.ce.ufl.edu/featured/tsis/	US	street data, simulated traffic flow

BonnMotion	Arbeitsgruppe Verteilte Systeme am Institut für Informatik der Universität Osnabrück	http://sys.cs.uos.de/bonnmotion/impressum.shtml	DE	mobility scenarios
Vissim	PTV Planung Transport Verkehr AG	https://www.ptvgroup.com/de/loesungen/produkte/ptv-vissim/	DE	street data, simulated traffic flow
VanetMobiSim	Eurécom	http://vanet.eurecom.fr	FR	street data, simulated traffic flow
SUMO (Simulation of Urban Mobility)	Institute of Transportation Systems of German Aerospace Center (DLR)	https://sumo.dlr.de/docs/index.html	DE	street data, simulated traffic flow
ASAM OpenDRIVE	ASAM e. V.	https://www.asam.net/standards/detail/opendrive/	DE	street data, simulated traffic flow

Table 8. External databases and traffic simulation solutions

5.2 Use Case Electricity Grid Operator (EGO)

5.2.1 Scenario Description

As introduced in chapter 3.1 the aim of this use case is to provide a third party, the electricity provider (EGO) with accurate and current weather data. This data is gathered by vehicle on the road within a specific area for which the EGO needs more, or more accurate information.

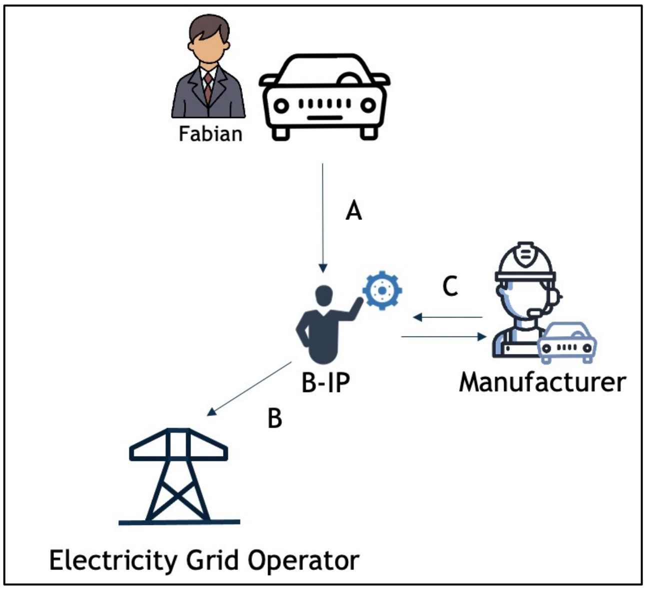


Figure 17: Data flow EGO

This use case comes with several *assumptions*:

- No direct personal data is shared about the owners and drivers of a vehicle
- EGO needs aggregated data once per minute for its purposes
- An optimal solution provides data privacy for all data types that occur in this use case

The entities in this use case are defines as follows:

- *Vehicle/owner:* The owner has a passive role in the data flow chart. The vehicle shares its sensor data with the B-IP. The concern of the owner is that no personal data about the owners and drivers of a vehicle is shared with the B-IP. The sensors of the vehicle create data in a frequency of 60 data points per minute.
- *B-IP:* The B-IP is responsible for analyzation and data preparation and follows the need-to-know principle. This means the B-IP only receives data that is mandatory to meet the requirements of the EGO.
- *EGO:* The EGO uses the data from the B-IP for energy demand predictions. Therefore, the EGO requires aggregated data once per minute. The data quality is required to exhibit enough information to make reliable energy demand predictions. Therefore, the EGO requires data in a frequency of 1 data point per minute.
- *Manufacturer:* The manufacturer only has a supporting role and is therefore not further considered in this scenario. No sensitive data is exchanged between B-IP and the manufacturer.

5.2.2 Data Flow Chart

In this section we establish privacy sensitivity and data gathering frequency for the different types of data that are to be used in the use case. The different communication channel are derived from Figure 17.

Communication Channel A: From owner/vehicle to B-IP

The following data is to be provided by the vehicles.

Data	Privacy Sensitivity	Data truthfulness at record level	Frequency
Brightness	Low	No	1/min
Rain	Low	No	1/min
Temperature	Low	No	1/min
Atmospheric pressure	Low	No	1/min
Humidity	Low	No	1/min
GPS	High	No	1/min
VIN	High	Yes	1/min

Table 9. Data types gathered by the vehicle

Communication Channel B: From B-IP to EGO

Both EGO and B-IP place requirements on the data quality and the frequency with which the data is to be provided to them.

Data	Privacy Sensitivity	Data truthfulness at record level	Frequency
Brightness	Low	No	1/min
Rain	Low	No	1/min
Temperature	Low	No	1/min
Atmospheric pressure	Low	No	1/min
Humidity	Low	No	1/min
GPS	High	No	1/min

Table 10. Data quality requirements by EGO and B-IP

Communication Channel C: From B-IP to Manufacturer

None

5.2.3 Analysis of De-Identification Techniques

Upon accessing the assumption and requirements of the electricity provider use case, all de-identification methods introduced in chapter 4.2. have been evaluated for their fit for the use case. Only de-identification methods that could initially demonstrate a sufficient level of privacy are discussed in detail below.

5.2.4 Attack Scenario

In this section we present possible attack scenarios. These form the basis for finding adequate solutions that can withstand such or similar attack scenarios.

Exact location determination

This attack tries to reveal the exact location of the vehicle/owner from a perturbated GPS location. This is done by combining the perturbated GPS location with brightness or rain data and an external database containing tunnel data. If all cars in a certain area report and one car does not, one can assume that this car was driving through a tunnel at the time of reporting. Because the number of tunnels in a certain area is limited, the location can be guessed precisely and thereby the perturbation is annulated.

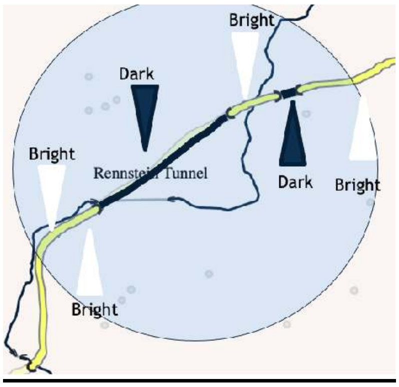


Figure 18. Exemplary vehicle route on map that contains tunnels

Steps

1. Vehicle sends data about brightness and GPS with an accuracy of 5 km every minute.
2. B-IP can access an external map with tunnel data.
3. The B-IP can reidentify cars by linking the tunnel data and the brightness data because in a certain area only a limited and known number of tunnels exists. If all other cars indicate daylight and one car indicates darkness, the car was in the tunnel at that time.

Vehicle Tracking and Track Localization

Even with perturbated GPS signals a malicious B-IP can easily track certain vehicles if the speed limits on certain roads are known. This information can easily be accessed with an external database. Although the B-IP does not get the true location, the average speed can be calculated over time and based on this possible roads or highways can be identified. Also, a database with traffic information containing traffic jams and accidents can leverage this attack.

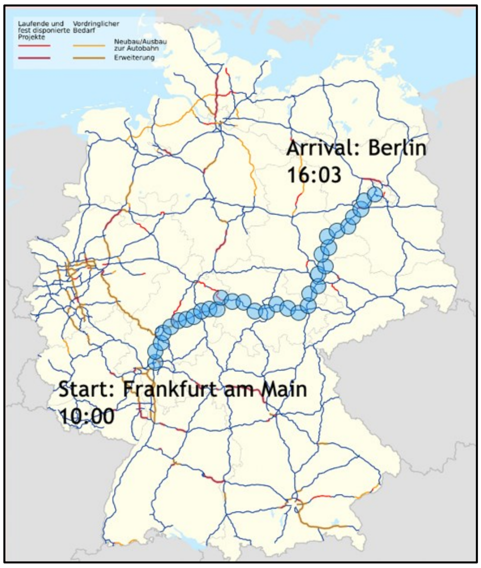


Figure 19. Examplary route from Frankfurt to Berlin whereby a vehicle provides location information periodically. Map: NordNordWest (2016), License:: Creative Commons by-sa-3.0 de, Bundesverkehrswegeplan 2030 Autobahnen.svg,

Steps:

1. Vehicle sends location to within 5 km, every minute to B-IP
2. B-IP calculates average speed based on approximate locations.
3. B-IP compares average speed with road network (highway/country road) and calculates route.

Linkability and Profiling

If a VIN number can be clearly mapped to a certain vehicle/owner a malicious B-IP can easily profile a certain vehicle/owner over time. Although data is sent perturbated and anonymized, the B-IP in this attack tries to identify certain vehicles and creates profiles over time.

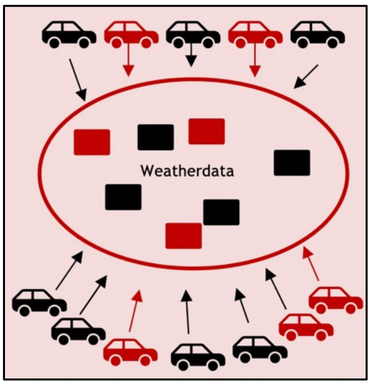


Figure 20. Vehicles provide weather information

Steps:

1. Vehicle sends weather data and GPS information anonymized.
2. B-IP tries to identify individual vehicles based on the anonymized weather data in combination with anonymized GPS location and with the help of external databases (e.g., external weather data).

5.2.5 Requirements for De-Identification

From the presented scenarios we derive the following requirements.

- Unlinkability: The B-IP should not be able to identify a certain owner/vehicle to lower the risk of profiling. This also holds for the EGO who should also not be able to identify a certain vehicle from the crowd.
- Location perturbation: No real GPS data is sent to decrease the risk of identification. This requirement becomes more difficult over time and is closely related to linkability.
- The quality of data should still be high enough to add value to the EGO's energy demand prediction model.

5.2.6 Analysis of De-identification Techniques

Upon accessing the assumption and requirements of the electricity grid operator use case, all de-identification methods introduced in chapter 4.2 have been evaluated for their fit for the use case. Only de-identification methods that could initially demonstrate a sufficient level of privacy are discussed in detail below.

Table 11 states methods that have been excluded as well as a brief statement on as to why they are deemed not suitable.

De-identification Technique	**Reason for exclusion**

Sampling	Sample data does not provide privacy protection for the specific sample and relies on a very high sample size which is likely not to be the case for vehicles driving in rural areas. In this use case, sampling cannot be used.
Deterministic encryption	Not suitable for weather data as only a very limited set of information will be used. This makes re-identification possible.
Aggregation	Aggregation might be usable for data that is gathered by multiple vehicles in a specific area but not for location information.
Order-preserving encryption	Not suitable as no ordered data is used.
Pseudonymization	Not suitable as a stand-alone solution. Not usable for weather data.
Generalization	Rounding or top/bottom coding not feasible for GPS data. No suitable upper and lower bound for weather information.
Randomization	Randomization alone does not protect against location tracking over time. Noise addition might be applicable but does only lead to acceptable results in combination with other techniques or models, e.g., K-anonymity.
Permutation	Reordering data does not work for a trajectory. The route of a vehicle could still be identified.

Table 11. Insufficient de-identification techniques for the electricity provider use case

The following de-identification techniques will be analyzed in the following:

1. Homomorphic Encryption
2. Secure Multiparty Computation
3. Distributed Differential Privacy
4. Federated Learning
5. K-anonymity

5.2.7 Approaches to De-Identification

Homomorphic Encryption

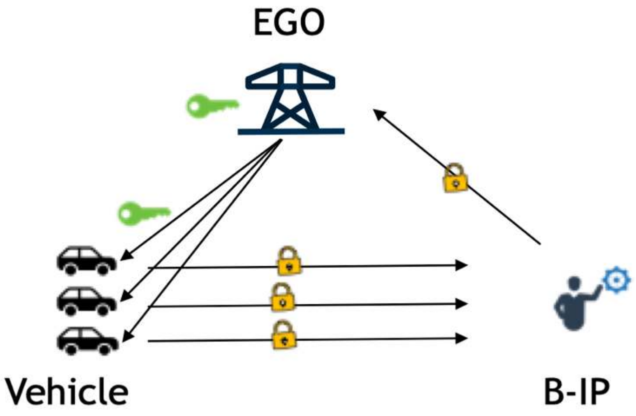
Diagram		
Entities	- *Vehicle:* Each car collects weather information on its own. - *Central Server (B-IP):* The B-IP acts as a central institution that collects encrypted data from the vehicles and computes operations on the encrypted data. - *Secret key:* The secret key is given to the vehicles by EGO. Vehicles use the key to encrypt the data that they then distribute to the B-IP. - *EGO:* Customer who wants to use the weather map to predict the power grid load.	
Steps	1. EGO distributed secret key to each vehicle. 2. Vehicle encrypts weather data homomorphically. 3. Vehicle sends the encrypted data to the B-IP. 4. B-IP computes the data, erased meta-data and sends the aggregated result to EGO. 5. EGO decrypts the data and is able to view the decrypted and aggregated results.	
Effort	Protective effect	High
	Complexity	High
	Runtime	High
	Degree of maturity	Medium
	Implementation effort	High
	Monetary cost	Medium
End result data quality	Time blur	Low
	Location blur	Low
	Processing speed	Low
	Aggregated data	Yes
	Truthfulness	Yes
	Time delay	Medium
Possible interfering factors	Communication overhead Network coverage	

Table 12. Evaluation of homomorphic encryption for the electricity provider use case

As explained in previous chapters the advantage of homomorphic encryption is that data can be computed while it is encrypted, guaranteeing that computations on the data lead to the same results on the decrypted data.

In the use case, the electricity providers distribute a private key to the vehicles on the road. The vehicles then use their key to homomorphically encrypt their location and weather data. The data is then distributed to the B-IP. The B-IP is now able to process the data as determined by the electricity provider. Meta data is deleted and average weather and location data is sent to the electricity provider. All these operations are performed on encrypted data, the BI-P is therefore unable to gain insights into vehicles actual locations and other provided information. However, operations on the decrypted data result in the same operation on the underlying data. The electricity provider is now able to use its secret key to decrypt the data and use the encrypted results for the intended purpose.

Nonetheless, homomorphic encryption creates several drawbacks. Although the technique itself has been available for some time, its actual usefulness is still hindered through the loss of performance and computational speed. Only a limited number of different operations, e.g., additions, subtractions can be computed, while run time increases greatly with the number of computations. However, research on homomorphic algorithms continues to improve run time, making homomorphic encryption a suitable solution for mobility-related use cases in the near future. Additionally, the techniques do not rely on the number of vehicles on the road and do not decrease the actual usability and truthfulness of the data.

Secure Multiparty Computation

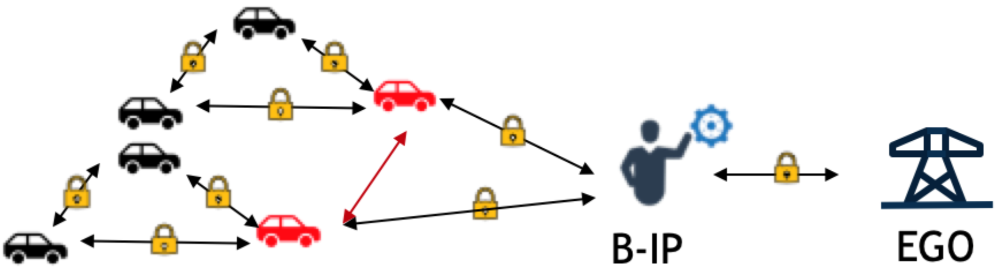
Diagram	**Secure Multiparty Computation**  	
Entities	• *Vehicle:* Each vehicle collects weather information and shares them through secrets with other vehicles • *Central Server (B-IP):* The B-IP collects only the end results per group of vehicles. • *EGO:* Customer who wants to use the weather map to predict the power grid load.	
Steps	1. Vehicles collect weather / brightness data. 2. Vehicle use MPC to create average brightness/ weather information in a specific region. 3. A group leader of the vehicles (red vehicle) collects the result of the MPC 4. Vehicle exchange group results and ID information through shuffling with other groups of vehicles. 5. Data is sent by a group leader to the B-IP. 6. B-IP may further process the data if necessary. 7. B-IP distributes data to EGO.	
Effort	Protective effect	High
	Complexity	High
	Runtime	High
	Degree of maturity	Medium
	Implementation effort	High
	Monetary cost	High
End result data quality	Time blur	Low
	Location blur	High
	Processing speed	Low
	Aggregated data	Yes
	Truthfulness	No
	Time delay	High
Possible interfering factors	Communication overhead	
	Network coverage	
	Vehicle density	

Table 13. Evaluation of Secure Multiparty Computation for the electricity provider use case

In this scenario we again assume a map that is separated in different cluster e.g., with a grid. Vehicles in each of these clusters calculate the average energy demand of a certain cluster with secure multiparty computation. One vehicle of each cluster is then chosen as the cluster leader that sends the computed result to the B-IP. To avoid an identification of certain vehicles and to reduce the size of necessary vehicles in a cluster, shuffling between the cluster leaders is also possible.

One approach for vehicular MPC communication is provided by Li et al. [42] who propose a cooperative control strategy incorporating with efficient MPC, reducing latency and integrating a function secret sharing scheme.

Interfering factors in this scenario are first, the vehicle density that is required per cluster. In case not enough vehicles are located in a certain cluster, no information can be calculated and sent to the B-IP. Second, a stable connection between the cars is required to use the MPC protocol. Third, the communication between the vehicles is likely to produce a huge overhead so that besides a good network coverage, a minimum bandwidth is mandatory.

K-anonymity

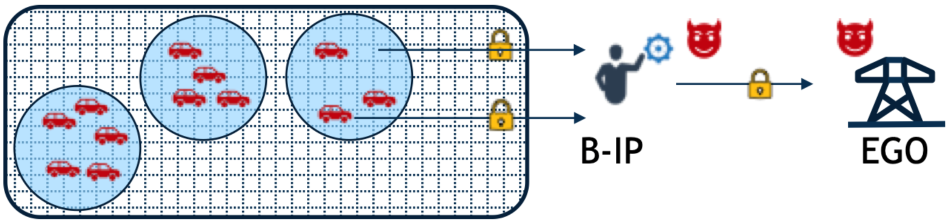
Diagram	**K-anonymity** 
Entities	• *Vehicle:* Each vehicle collects weather information and uses different techniques to hide its identity and data against possible re-identification. • *Mix-point:* Within a mix point, vehicle use different techniques and protocols to hide their information. • *Central Server (B-IP):* The B-IP receives information from the vehicles to compute aggregated results. • *EGO:* Customer who wants to use the weather map to predict the power grid load.
Steps	1. Vehicles collect weather / brightness data. 2. Vehicles get assigned to a mixing point that satisfies the k-anonymity property given a specific k. 3. Within a mixing point, vehicles use different techniques such as hiding, cloaking, dummyfiying to create k-anonymity. 4. Depending on the protocols used, vehicles send aggregated or fixed location data of the vehicle or mixing point. Vehicles send information to the B-IP. 5. B-IP computes aggregated data and sends it to EGO. 6. EGO receives aggregated data to plan power grid load.
Effort	Protective effect: Medium Complexity: Low Runtime: Medium Degree of maturity: High Implementation effort: Low Monetary cost: Low
End result data quality	Time blur: Medium Location blur: High Processing speed: Medium Aggregated data: Yes Truthfulness: No Time delay: Medium
Possible interfering factors	Communication overhead Network coverage Vehicle density Topology

Table 14. Evaluation of K-anonymity for the electricity provider use case

K-anonymity in itself is not a de-identification technique but a property with which data privacy in a database might be measured. ISO/IEC 20889_2018 defines K-anonymity as a formal privacy measurement model which ensured that an equivalence class in a database contains at least K records that are similar for each identifier. K may be chosen by the owner or user of the database whereby a higher K ensures a higher level of privacy. A specific record is similar to the number of K other records. Several enhancements, e.g., L-diversity and T-closeness of K-anonymity, exist that mitigate some flaws in the K-anonymity model.
A significant amount of research has been conducted on mobility related use cases such as VANETs using K-anonymity as a measurement model to ensure privacy of vehicles.

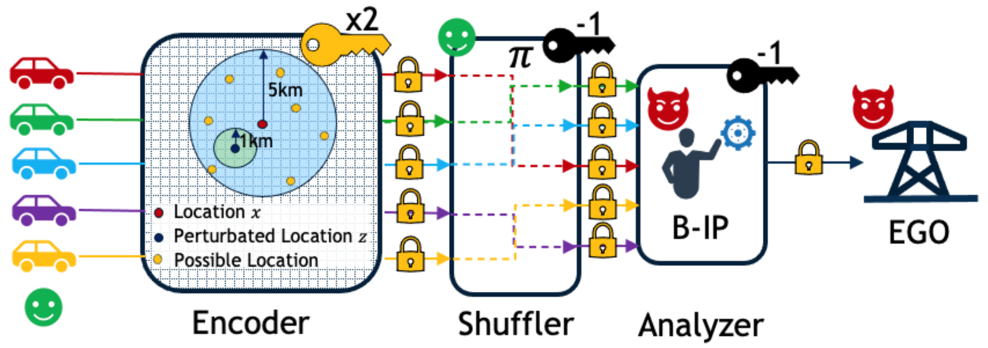
In this specific use case, the objective of vehicles is to obfuscate their exact location and ensure that weather information cannot be used for location inference.

For K-anonymity, a map is clustered into various mix points whereby each mix point fulfils the K-anonymity requirement. In the electricity provider use case, the map represents the area in which vehicles are to gather weather information. This area is divided into mix points to increase the accuracy of information. The work of [17] introduces multiple different protocols to create mix points such as stationary mix points, mix points occurring at irregular time intervals or randomly chosen mix points that may occur regularly or irregularly. An additional option would be that vehicles themselves create mix points and act as group leaders of other vehicles, thereby managing the fulfillment of K-anonymity and the data distribution behavior of a group of vehicles. Within such a mix point, whose center may for instance be an intersection, vehicles switch pseudoIDs with other vehicles and/or are added to an anonymity set and do not communicate information for a specific time period. Essentially, the work uses the de-identification techniques of suppression and pseudonymization to achieve K-anonymity. Additionally, such a model could be enhanced by adding further simple de-identification techniques such as aggregation, noise addition or permutation to it. Such options would enhance privacy at the cost of a loss of quality of service as the usefulness of data decreases. In [17] the authors decided against the use of such options as anonymizing, for instance through spatial cloaking, cannot effectively protect against tracking over time and leads to less precise results. Dummifying has not been used as false location data might lead to accidents as the authors use case has been to provide relevant safety traffic data to other vehicles through a central service. However, in our use case, exact location data is not as important as in other use cases as the weather might not differ strongly in a 500m radius. Time delay might also be acceptable, to an extent, as weather will not change significantly within 5 minutes.

Therefore, a combination of simple de-identification techniques that fulfill K-anonymity are seen as a suitable alternative for the electricity provider use case. In any case, the protective effect of this solution will not be as high as that as more advanced methods such as MPC. Multiple factors affect the level of privacy that can be obtained: A lower vehicle density results in a lower K-value and a lower level of privacy. The topology, e.g., the number of roads and the speed of travel, influence privacy as fewer roads lead to less privacy. Similarly, the choice and design of mixing points, depending on the chosen protocol, need to be matched with such factors.
Complexity of the model is low while the runtime again depends on the choice of techniques

and protocols to be used. Such protocols however already exist, creating mature solutions that could be implemented quickly and at low monetary cost. As has been elaborated, data may be sent from each vehicle or aggregated between vehicles. Data could include dummy variables, resulting in non-truthful data. Depending on the number of vehicles in a mix point and on the road, the usefulness of the data might change. Less vehicles equal larger mixing points and an increase in location blur and possibly time delay in order to ensure privacy. Overall, while K-anonymity-based solutions might provide a cheap solution that can be implemented easily, data quality and the achievable level of privacy greatly depend on topology and the number of vehicles within an area.

Distributed Differential Privacy

<table>
<tr><td><b>Diagram</b></td><td colspan="2">
</td></tr>
<tr><td><b>Entities</b></td><td colspan="2">• <i>Encoder (Car):</i> Each car has an encoder that can encode data and permute GPS locations before sending.
• <i>Shuffler:</i> An additional trusted entity which performs data anonymization, shuffling, thresholding and batching.
• <i>Analyzer (B-IP):</i> Sole principal of data which performs decryption, data storing and data aggregation.
• <i>EGO:</i> Customer who wants to use the weather map to predict the power grid load.</td></tr>
<tr><td><b>Steps</b></td><td colspan="2">1. The <i>Encoder (Car)</i> perturbates the location and creates an inner and outer encryption before sending the data to the <i>Shuffler</i>. This happens every minute.
2. The <i>Shuffler</i> decrypts the outer layer, strips the metadata and uses shuffling, thresholding and batching to anonymize the data before forwarding the batch to the Analyzer. The batches are sent on an irregular basis but the data in a batch cannot be older than 5 minutes.
3. The <i>Analyzer (B-IP)</i> decrypts the inner encryption and uses the data to create a weather map that is sent to the EGO.
4. The EGO receives weather maps to make energy forecasts and adjust the peak load in a certain area if necessary.</td></tr>
<tr><td rowspan="6"><b>Effort</b></td><td>Protective effect</td><td>High</td></tr>
<tr><td>Complexity</td><td>High</td></tr>
<tr><td>Runtime</td><td>High</td></tr>
<tr><td>Degree of maturity</td><td>High</td></tr>
<tr><td>Implementation effort</td><td>High</td></tr>
<tr><td>Monetary cost</td><td>High</td></tr>
<tr><td rowspan="4"><b>End result data quality</b></td><td>Time blur</td><td>Medium</td></tr>
<tr><td>Location blur</td><td>Medium</td></tr>
<tr><td>Processing speed</td><td>High</td></tr>
<tr><td>Aggregated data</td><td>Yes</td></tr>
</table>

	Truthfulness	Yes (No for GPS)
	Time delay	Medium
Possible interfering factors	Car density Area topology Car speed	

In this scenario description we utilize the system architecture Encode, Shuffle, Analyse (ESA) proposed by Bittau et al., 2017 [43] to implement distributed differential privacy. In general, the architecture consists of three entities, an encoder, a shuffler and an analyzer, as seen above. In the following we will have a detailed look at the tasks of each entity in our concrete scenario with the EGO.

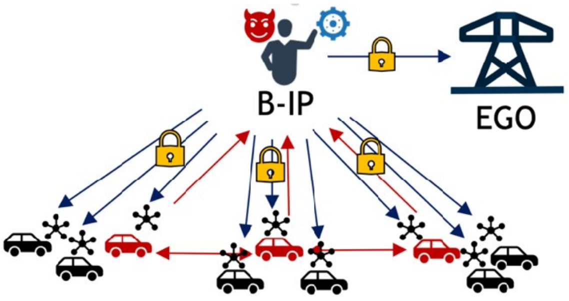
Encoder: The encoder is responsible for ensuring the fulfilment of the user's trust assumptions by locally transforming and conditioning the user's private data [43]. In our EGO use case one of these transformations is the location perturbation providing local differential privacy as proposed by Andrés et al. [44] by fulfilling the requirement of geo-indistinguishability. Moreover, the encoder is responsible for the encryption of the data with an inner and outer encryption, and the transmission over a secure channel to the shuffler. As explained above, the encoder entity is placed on the user's device, in the EGO use case, we place the encoder in the car.

Shuffler: The shuffler acts as an additional privacy layer in between the user's encoder and the analyzer that should be run by a trusted third party. The shuffler is responsible for the anonymization, shuffling, thresholding, and batching of the data received from the encoder. By decrypting the outer encryption, the shuffler can access the metadata of a user, e.g., timestamps, source IP addresses, routing paths. The main task of the shuffler is to remove all this data before forwarding it to the analyzer. To prevent the reassignment of the data by the analyzer to a certain user, the data are reordered randomly and forwarded infrequently and only in batches. Moreover, the shuffler can also set thresholds and reject data items to ensure that each item can be hidden in a sufficient crowd.

Analyzer: The analyzer is responsible for the innermost decryption, storing and aggregation of the data received from the shuffler. The analyzer utilizes techniques such as differential privacy to make the data available for other groups of interest without revealing private user information. In the EGO use case this role is taken by the B-IP. The B-IP uses the data received from the shuffler to create a weather map that is sent to the EGO.

The biggest issue of this approach is car density and appears if only viewed cars are in a certain location. As a result of this, a single car cannot be hidden sufficiently in the crowd and the shuffler has to delay or withdrawal the forwarding of certain batches. Therefore, a minimum number of cars per region is required. Moreover, the number of cars is influenced by area topology and daytime. In a scenario where the EGO wants to make assumptions on the required network load, e.g., for vehicular charging, the absence of data in a certain region would point to a very low electricity demand. The average demand for an area could be set approximately on historic results or in dependency of the minimum number of cars.

Federated Learning

Diagram		
Entities	• *Vehicle:* Each *vehicle* collects weather information and can share information with other cars within a certain radius. • *Cluster:* Several vehicles within a certain radius or region form a cluster. Each cluster has a minimum size and exactly one *Leader.* • *Leader:* The leader is the vehicle https://meteostat.net/dein a cluster that sends updates to the B-IP if the locally stored model parameters have changed. • *Central Server (B-IP):* The B-IP acts as central server. The central server exhibits the central model that is updated by the Local information of the *Leaders.* After one round of updating the central server distributes the new model to all cars. • *Model:* The model contains the weather information of every region. Each *Car* receives the central model from the *B-IP* and stores it locally. • *EGO:* Customer who wants to use the weather map to predict the power grid load.	
Steps	1. The *B-IP* distributes a pretrained model to all cars. 2. Each *Car* receives the central model and stores it as local model. 3. If a weather parameter in the local model is changed, each vehicle updates the local model. 4. The *B-IP* requests a parameter update every minute 5. A cluster determines a *Leader* that shuffles the data with all other leaders and then sends the data to the *B-IP.* A *Leader* only participates in shuffling if a parameter has changed in the cluster.	
Effort	Protective effect	High
	Complexity	High
	Runtime	High
	Degree of maturity	Medium
	Implementation effort	High
	Monetary cost	High
End result data quality	Time blur	Low
	Location blur	Low
	Processing speed	High
	Aggregated data	Yes
	Truthfulness	Yes
	Time delay	Medium
	Car density	

Possible interfering factors	Communication overhead Network coverage

Saputra et al. [45] propose a federated learning model for energy demand prediction for electric vehicle networks but compared to our approach they utilize the information gathered from the charging stations. On the one hand, they use a federated learning model with the aim to reduce the communication overhead between the charging stations and the main server with the central server. On the other hand, they protect the data of the vehicle users by only transmitting relevant information in the form of parameter updates to the central server rather than sending whole data sets.

Liu et al. [27] present a traffic flow prediction scheme using location-based clustering in combination with a federated learning approach. In their approach they collect the information from organizations (e.g., bus stop or station) while randomly selecting only a defined ratio of organizations from a larger group in each round of training. Yin et al. [46] propose a Federated Localization (FedLoc) framework with the aim to build accurate location services without revealing sensitive user information. They propose a cloud-based network infrastructure that is based on many clusters that do not overlap. These clusters are defined by the mobile communication range of a mobile terminal, e.g., 5G macro and micro base stations and WiFi6-networks that can enable a high communication rate.

As explained above, the federated learning scenario requires a clustering of vehicles that communicate with a central server. This could be solved by the cars communication radius with other cars or as proposed by Yin et al. [46] by utilizing the radius of base stations. In both cases, the cars will have to communicate with each other to determine a Leader of each cluster to communicate in a certain training round with the B-IP. The cars in a certain cluster can either calculate an average of their collected data by using MPC or just blur the exact data location data with e.g., location perturbation. In case the weather parameters in a certain cluster did not change, the leader will not participate in the current round of training to keep the traffic as low as possible. To fulfill the requirement of unlikability, a distributed shuffling protocol as proposed by Cheu et al. [47] between all Leaders of a cluster can be used to delete metadata and shuffle the data between the Leaders. The leaders then send an updated model to the B-IP. The B-IP cannot link the received data to the sender because the data was shuffled before and metadata was deleted. In each training round, e.g., every minute, the B-IP receives updated models from the leaders. These models are then used to develop a new central model. This model is then sent to the EGO and also distributed to all cars. A possible extension to keep the traffic low is to determine the new Leader for a certain round in advance and only use the Leader's data in that round. The Leader could still exchange data with other cars in the cluster but the model is only with the Leader. In the future further experiments will be required to build the most efficient model.

The biggest pitfalls for the Federated Learning approach are car density and network coverage. A minimum number of vehicles is required to form a cluster, otherwise, no information can be sent to the B-IP. Moreover, a lot of communication is required for this distributed learning approach, therefore a sufficient network coverage is mandatory.

5.2.8 Comparison of Technologies

De-Identification Techniques for use case electricity provider	**Homomorphic Encryption**	**Secure Multiparty Computation**	**Distributed-ted Differential Privacy**	**Federated Learning**	**K-anonymity**
Protective effect	High	High	High	High	Medium
Complexity	High	High	High	High	Low
Runtime	High	High	High	High	Medium
Degree of maturity	Medium	Medium	High	Medium	High
Implementation effort	High	High	High	High	Low
Monetary cost	Medium	High	High	High	Low
Data quality					
Time blur	Low	Low	High	Low	Medium
Location blur	Low	High	Medium	Low	High
Processing speed	Low	Low	Medium	High	Medium
Aggregated data	Yes	Yes	Yes	Yes	Yes
Truthfulness	Yes	No	Yes (No for GPS)	Yes	No
Time delay	Medium	High	Medium	Medium	Medium
Possible interfering factors	Communication overhead Network coverage	Communication overhead Network coverage Vehicle density	Car density Area topology Car speed	Car density Communication overhead Network coverage	Network coverage Vehicle density Topology

Table 15. Aggregated results

When comparing the different solutions for the electronic grid operator use case, we find that all advanced de-identification techniques are able to provide a high level of privacy for individuals and vehicles. However, all solutions are relatively complex and most of them require further research. While a solution based on k-anonymity offers the least amount of privacy protection, it is easily implementable, cheap with an acceptable data output. Distributed differential privacy and federated learning are both more complex solutions that provide more fine-grained insights as data quality remains higher. External factors such as vehicle density, travelling speed and network coverage will likely greatly influence the successful application of each use case. Here, traffic flow simulations could be used to verify solutions by combining simulated traffic scenarios with actual vehicle data.

5.3 Use Case Pedestrian

5.3.1 Scenario Description

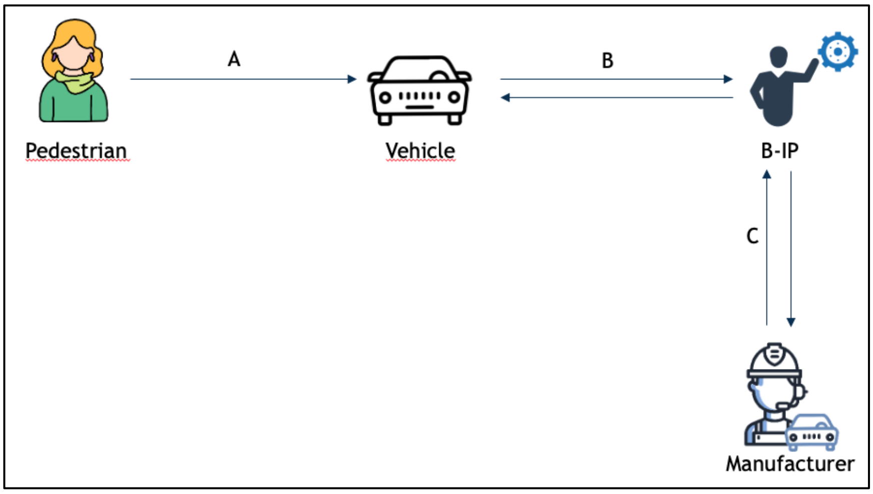


Figure 21: Flow chart bystanders

In this use case an autonomous driving vehicle is equipped with a camera and driving on a public road. The aim is to detect the line of sight of uninvolved pedestrians. Here, the direction of gaze is to be detected for motion prediction of pedestrians. This serves the early detection and avoidance of dangers in road traffic.

For this use case we assume that identification of pedestrians is not possible using Lidar or Radar sensors. Identification is only possible using cameras through 3D images. The main goal of this use case is therefore to retain information on the direction in which a pedestrian is looking and walking, while anonymizing the pedestrian's biometric data. Personal data is not to be sent to any other entity from the vehicle.

- *Pedestrian/Environment:* We define the pedestrian as a random person who is captured by a camera from a vehicle, whether the vehicle is stationary or driving is neglected. The pedestrian does not want to be identified by the vehicle. Therefore, features that would identify them must be made unrecognizable. Simultaneously, other vehicles are driving on the road that depict license plate information.
- *Vehicle:* For autonomous driving support systems such as automatic braking are required. To evaluate a situation where a pedestrian is moving close to the close to the roadway the direction of movement must be determined. Therefore, the vehicle is equipped with a camera and a machine learning based model that can predict the direction a pedestrian is moving. The vehicle can share information with the B-IP to improve the model.
- *B-IP:* The B-IP is responsible to maintain, train and update the model that is used e.g., for movement prediction. The B-IP can receive vehicle specific information from the manufacturer. The B-IP continuously communicates with the vehicle. The B-IP also creates and sends reports to the manufacturer. These reports contain information about certain vehicle models but not on a certain vehicle.

- *Manufacturer:* The manufacturer has a supporting role and provides vehicle specific information to the B-IP. The manufacturer also receives reports about the status of certain vehicle models. The manufacturer validates updates from the B-IP.

5.3.2 Data Flow Chart

In this section we will go through each communication channel in Figure 21.

Communication channel A: From Pedestrian to Vehicle

Data	Privacy Sensitivity	Data truthfulness at record level	Frequency
Environmental	Medium	Yes	1/ms
Biometric	Critical	Yes	1/ms
License plate number	Critical	Yes	1/ms
Distance	Uncritical	Yes	1/ms
Timer	Uncritical	Yes	1/ms

Communication channel B: Vehicle to B-IP

Data	Privacy Sensitivity	Frequency
Model updates	Medium	1/month
Model failures	Medium	If needed

Communication channel B: B-IP to Vehicle

Data	Privacy Sensitivity	Frequency
Model updates	Medium	1/month

Communication channel B: B-IP to Manufacturer

Data	Privacy Sensitivity	Frequency
Report	Medium	1/month

Communication channel B: Manufacturer to B-IP

Data	Privacy Sensitivity	Frequency

Model updater request	Medium	1/month

5.3.3 Trust assumptions

- *Pedestrian:* Untrustworthy
- *Vehicle/Owner:* Untrustworthy
- *B-IP:* Untrustworthy
- *Manufacturer:* Trustworthy

5.3.4 Attack Scenarios

Location Determination

The location from each person that is filmed while driving can be easily set by comparing the timestamp of the GPS location of the vehicle with the timestamp of the photo.

Steps:

1. Create videos or pictures of pedestrians while driving.
2. Link the timestamp of the picture/video to the timestamp of the vehicle's GPS locations.
3. Create a database of pedestrians and GPS locations that can be enriched using external databases that contain street and location data as well as social media information.

Identification

The vehicle/owner can use a face recognition algorithm to identify a person. This algorithm could be trained on an additional database that maps profile pictures and names. Nowadays, most people have a profile picture available on social media, professional networking platforms or on their companies' website that can be easily collected by third parties. In a further step, location profiles could be created.

Steps:

1. Collect data bases with names and pictures of people.
2. Train a face recognition model.
3. Identify people by name.

Motion Profile Creation

In this scenario the vehicle/owner collects and stores pictures or short videos of people that are filmed while driving. The focus of this attack is on the clustering of similar pictures or videos from the collected data. The model can be trained so that a cluster only contains only photos of a particular person. In combination with the location of the vehicle, at the time a picture was made a location can be mapped to each picture/video, as described above. Based on this, a motion profile can be created. This attack can be expanded with the identification attack, adding more private information about a person.

Steps:

1. Collect pictures or videos of pedestrians and store them in a database.
2. Use the location data from the vehicle to assign each picture/video in the database a location.
3. Cluster the collected data so that one cluster can be assumed to be one person.
4. Combine location data and clustering results to create motion profiles.

5.3.5 Requirements for Approaches to De-Identification

- B-IP does not receive personal data from the vehicle.
- PII (e.g., biometrics and license plate data) are not saved in the vehicle.
- Vehicle Manufacturer does not get PII from the vehicle or the B-IP.

5.3.6 Analysis of De-identification Techniques

Upon accessing the assumption and requirements of the personalized services use case, all de-identification methods introduced in chapter 4.2 have been evaluated for their fit for the use case. Only de-identification methods that could initially demonstrate a sufficient level of privacy are discussed in detail below.

Table 16 states methods that have been excluded as well as a brief statement on as to why they are deemed not suitable.

De-identification Technique	**Reason for exclusion**
Sampling	Sample data does not provide privacy protection for the specific sample and relies on a very high sample size which is likely not to be the case. In this use case, sampling cannot be used.
Aggregation	Aggregation might be usable at the pixel level for biometric data as a first step but cannot provide sufficient protection in this use case.
Order-preserving encryption	Not suitable as no ordered data is used.
Homomorphic encryption	Mathematical operations too complex for homomorphic encryption in this use case.

Secure Multiparty Computation	Not suitable as no computations over multiple entities are necessary.
Pseudonymization	Not suitable.
Randomization	Noise addition might be applicable for biometrics but is highly likely to decrease the usability of the data to an undesired degree.

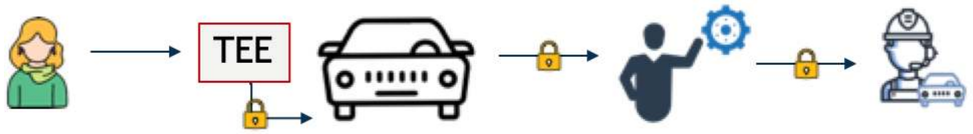
Table 16. Insufficient de-identification techniques for the electricity provider use case

The following de-identification techniques will be analyzed in the following:

1. Federated Learning
2. Trusted Execution Environment (TEE)
3. Differential Privacy

5.3.7 Approaches to De-identifications

Trusted Execution Environment (TEE)

Diagram	**Trusted Execution Environment (TEE) using masking**  	
Entities	• *Vehicle:* A vehicle collects environment and biometric pedestrian data while driving autonomously. A TEE within the vehicle is used to store the data safely and run the autonomous driving algorithm. • *Central Server (B-IP):* The B-IP receives analysis results of the data from the vehicle in order to improve its existing autonomous driving algorithm. • *Manufacturer:* Only has a supporting role. • *Pedestrian:* A pedestrian is walking on a public street next to the road on which the vehicle is driving. The pedestrian is not aware that the vehicle is recording them.	
Steps	1. Vehicles collect biometrics and environment data while driving. 2. Vehicle uses masking, blurring and obfuscation, or Generative Adversarial Networks (GANs) to anonymize the data. This is done in a TEE within the vehicle. 3. Results are continuously sent to the B-IP. 4. The B-IP uses the non-personal data to train its model. Model improvements are sent back to the vehicle. 5. The B-IP sends periodic status reports to the manufacturer.	
Effort	Protective effect	High
	Complexity	Medium
	Runtime	Medium
	Degree of maturity	High
	Implementation effort	Medium
	Monetary cost	Medium
End result data quality	Time blur	Low
	Location blur	Medium - High
	Processing speed	High
	Aggregated data	Yes
	Truthfulness	No
	Time delay	Low
Possible interfering factors	Vehicle processing capabilities Pedestrian density and background Pedestrian poses Occlusion	

In this solution, data is analyzed in a TEE within the vehicle and anonymized using masking and obfuscation. A wide variety of algorithms and models to anonymize biometric data exists. Solutions that rely on pixelation or blurring are deemed unusable for this use case as it is necessary that the data is used for further training the autonomous driving algorithms. Therefore, the data distribution should not be altered significantly. Additionally, while individuals should be anonymized, important information such as the direction in which a pedestrian is looking, should be maintained. Other information, such as facial attributes, is not important for our use case, making solutions that retain such information, while still anonymizing the individual, unnecessarily complex. Thus, a suitable solution should only have to retain the direction in which a pedestrian is looking.

One example is the work of [48] in which the authors propose DeepPrivacy, a model that automatically anonymizes faces. The model uses a conditional generative adversarial network (cGAN) that generates new images that are similar to the existing faces, to model natural image distribution. The broad steps and results of the model can be seen in Figure 22. The authors show that such a solution offers a high level of privacy and demonstrate that the solution achieves higher privacy than blurring, pixelation or blacking out parts of an image. As a drawback, it is shown that unrealistic results, such as blurry faces, occur in case of "high occlusion, difficult background information, and irregular poses" [48]. A more recent improvement of the model is also able to replace license plate information and biometric data in videos.[4]

Although the image generator model does not see any personal data, such a solution could be run in a TEE within the vehicle. Thus, personal data would not be stored in the vehicle and could not be transferred to other entities such as the B-IP. The vehicle would then anonymize the data and only transfer anonymized data using GANs to the B-IP for improvements on the model. Similarly, the vehicle manufacturer would not receive personal data.

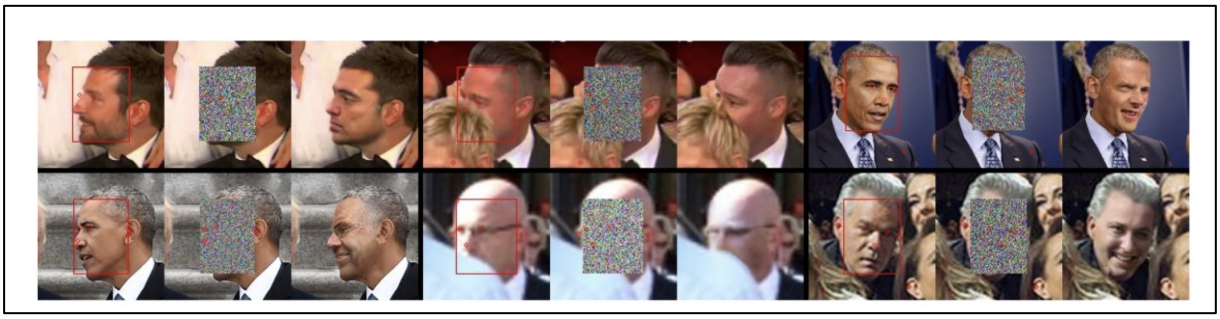


Figure 22. DeepPrivacy results [48]: Left picture depicts the original image, middle picture the input into the network and right image the newly generated image.

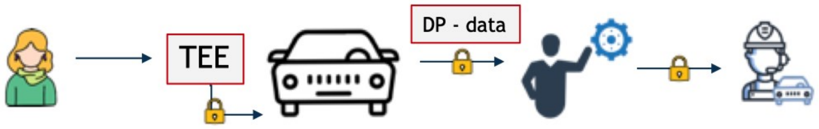
[4] For more information, see: Burt, 2019. New anonymization solution from D-ID blocks facial biometrics with fake faces https://www.biometricupdate.com/201908/new-anonymization-solution-from-d-id-blocks-facial-biometrics-with-fake-faces (Last visited: 29.01.2021)

Other methods exist that are also usable for video data, leading to similar results.[5]

Overall, it can be seen that a TEE provides an ideal environment to anonymize biometric data generated through cameras in a vehicle. TEEs are already being used to store biometric data, e.g., in smartphones using face recognition. In a vehicle, models such as a conditional generative adversarial network could be run in a TEE to ensure that all collected biometric data is immediately anonymized. The respective technology already exists and generated a sufficient degree of privacy for its actual implementation and usage. Runtime of such a solution is rated as medium as today's models are already able to perform anonymization in real-time, for instance, in smartphone applications. Implementation and monetary effort are rated as medium as a dedicated TEE would need to be installed in each vehicle, while the implementation of the anonymization model is seen as low. The solution is highly useful as all relevant information can still be obtained from the data in near to real-time. However, the anonymization model might have difficulties with a high density of pedestrians in an area, or if faces are partially covered by other objects, making continuous improvements of the model necessary.

[5] For a comparison of DeepPrivacy and another model called CLEANIR, see: Nagaraj, 2020. Face Anonymization: A survey of what works and what doesn't. https://blog.ml6.eu/face-anonymization-a-comparison-66da5088d030 (Last visited: 29.01.2021)

Differential Privacy and TEE

Diagram	**Differential Privacy and TEE**  	
Entities	• *Vehicle:* A vehicle collects environment and biometric pedestrian data while driving autonomously. A TEE within the vehicle is used to safely store the data and run the autonomous driving algorithm. • *Central Server (B-IP):* The B-IP receives analysis results of the data from the vehicle in order to improve its existing autonomous driving algorithm. • *Manufacturer:* Only has a supporting role. • *Pedestrian:* A pedestrian is walking on a public street next to the road on which the vehicle is driving. The pedestrian is not aware that the vehicle is recording them	
Steps	1. Vehicles collect biometrics and environment data while driving. 2. Vehicle uses masking, blurring and obfuscation, or Generative Adversarial Networks (GANs) to anonymize the data. This is done in a TEE within the vehicle. 3. Results are continuously sent to the B-IP. 4. The B-IP uses the non-personal data to train its model. Model improvements are sent back to the vehicle. 5. The B-IP sends periodic status reports to the manufacturer.	
Effort	Protective effect	High
	Complexity	High
	Runtime	Medium
	Degree of maturity	Medium
	Implementation effort	Medium
	Monetary cost	Medium
End result data quality	Time blur	Low
	Location blur	Medium - High
	Processing speed	Low - Medium
	Aggregated data	Yes
	Truthfulness	No
	Time delay	Medium
Possible interfering factors	Communication overhead Vehicle processing capabilities Pedestrian density Choice of training data Camera sensor quality	

To introduce differential privacy in this use case, we built upon existing work on privacy-preservation of biometric data. The authors of [49] introduce a protocol for privacy-preserving face recognition that utilizes local differential privacy for perturbation of the data. The exemplary model and its functioning can be seen in Figure 23.

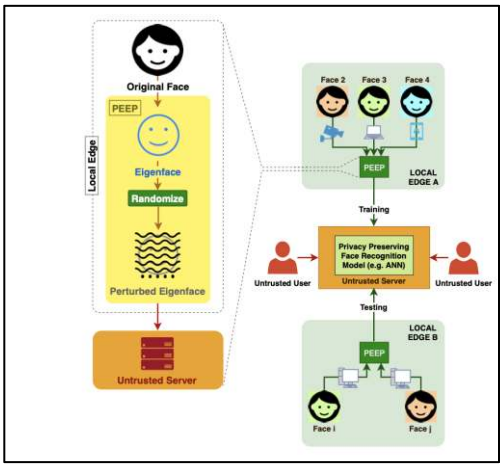


Figure 23. Example of Differential Privacy Model PEEP by [49]

Here, the model is placed in a facial recognition system and both training and testing data are randomized. The model first accepts the original facial data, generates eigenfaces, a set of eigenvectors. Then, Laplacian noise is added to the eigenface to randomize the result. The authors find that their solution is scalable, data is not linkable to other sensitive data and biometrics are not accessible for third parties. Facial recognition time has a complexity of O (1). The performance time can be seen in Figure 24 given the hardware used by the authors. The privacy budget ε, which gives an insight into the loss of privacy that a differential privacy algorithm creates, is set to 8, a generally accepted level of privacy loss in the data.

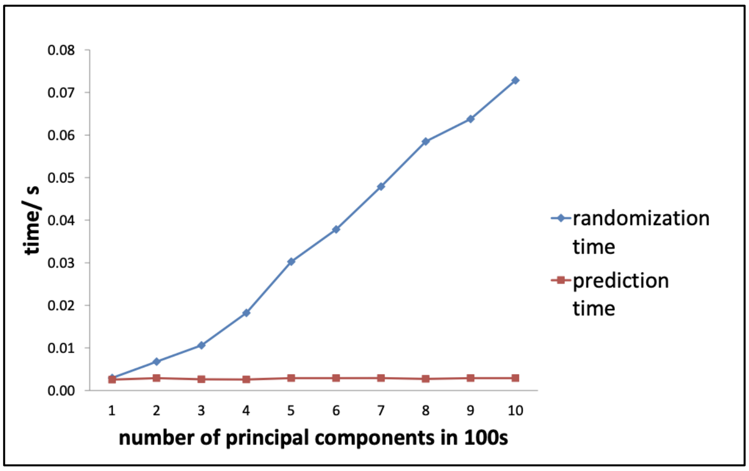
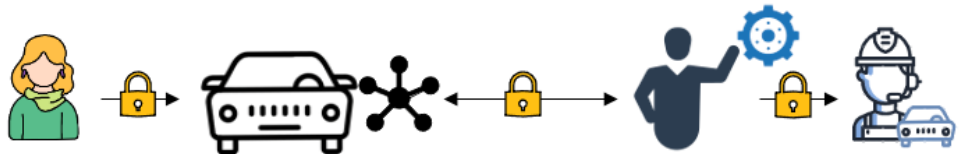


Figure 24. Performance of the PEEP model for face recognition and randomization

We can adapt this protocol for the pedestrian use case. Upon applying perturbation on the data, only the perturbed data would then be stored on the insecure hardware of the vehicle or is sent to untrusted servers belonging to the B-IP. Alternatively, the data could also be stored and perturbed on a TEE in the vehicle. The vehicle would then use the anonymized biometric data for its autonomous driving algorithms. Reports are sent to the B-IP. The solution offers a high level of privacy, as shown through the differential privacy metric. Its complexity is listed as high while the runtime is listed as medium. While the identification of faces is not an issue, the time to perturb the data increases greatly with the amount of data that is to be anonymized. Here, pedestrian density, their poses and occlusion might decrease the runtime significantly. As can be seen, multiple such solutions exist for biometric data, demonstrating that perturbation is a viable and cheap solution for this use case. However, multiple factors influence the efficiency of this solution. The number of people and objects that need to be anonymized may increase randomization time while camera sensors require and TEE require additional hardware within each vehicle.

Federated Learning

Diagram	**Federated Learning** 	
Entities	- *Vehicle:* A vehicle collects environment and biometric pedestrian data while driving autonomously. Each vehicle exhibits its own local model where the computation takes place. - *Central Server (B-IP):* The B-IP owns the central model and can request gradient updates from the local models. - *Manufacturer:* Only has a supporting role. - *Pedestrian:* A pedestrian is walking on a public street next to the road on which the vehicle is driving. The pedestrian is not aware that the vehicle is recording them.	
Steps	6. Vehicles collect biometrics and environment data while driving. 7. Vehicle uses masking, blurring and obfuscation, or Generative Adversarial Networks (GANs) to anonymize the data. This is done within the local model on the vehicle. 8. Results stay locally on the vehicle. 9. The B-IP can request gradient updates and send model improvements back to the vehicle. 10. The B-IP sends periodic status reports to the manufacturer.	
Effort	Protective effect	High
	Complexity	High
	Runtime	Medium
	Degree of maturity	Medium
	Implementation effort	High
	Monetary cost	High
End result data quality	Time blur	Low
	Location blur	Medium
	Processing speed	High
	Aggregated data	Yes
	Truthfulness	No
	Time delay	Low
Possible interfering factors	Vehicle processing capabilities Pedestrian density and background Pedestrian poses Communication overhead	

One benefit of federated learning is the local processing of data in the vehicles local model. Thereby, an aggregation of data is avoided and only gradient updates are shared with the B-IP. These updates contain e.g., weights of a neuronal network but no privacy sensitive data.

Although the local processing can ensure that data stays in the vehicle, the collection of privacy sensitive data through sensors is not prevented. This is the reason why in this scenario further technologies, such as blurring and obfuscation, or Generative Adversarial Networks (GANs) are required to anonymize the data collected by the sensors before they are computed by the local model.

Federated learning for object categorization with street data is described by Luo et al. [50]. Their model is built on images generated by several street cameras. They line out the advantage of federated learning to build robust models on highly imbalanced data. A possible federated learning based architecture for vehicular networks is also provided by Elbir et al. [51]. Their federated learning updating scheme is shown in Figure 25.

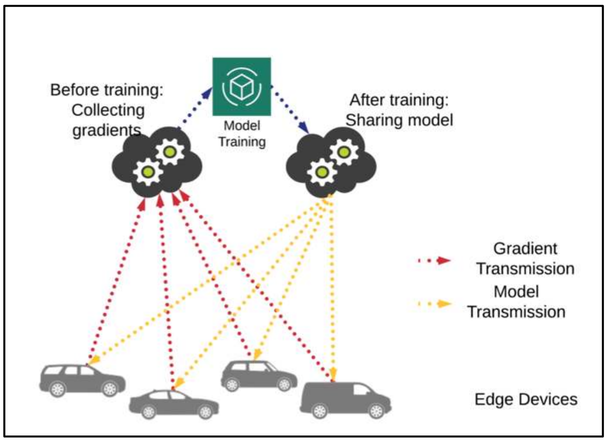


Figure 25: Vehicular federated learning network [51]

Possible interfering factors in this scenario are transmission overhead but also security issues by malicious participants. Therefore, a reputation management of devices participating in the federated learning scenario is required. Moreover, general issues to pedestrian recognition are pedestrian density, background noise and unfamiliar poses. In general, federated learning is a highly recommended approach to tackle these issues.

5.3.8 Comparison of Technologies

De-identification techniques for use case pedestrian	Trusted Execution Environment (TEE)	Differential Privacy with TEE	Federated Learning
Protective effect	High	High	High
Complexity	Medium	High	High
Runtime	Medium	Medium	Medium
Degree of maturity	High	Medium	Medium
Implementation effort	Medium	Medium	High
Monetary cost	Medium	Medium	High
Data Quality			
Location blur	Medium - High	Low	Low
Processing speed	High	Medium	High
Aggregated data	Yes	Yes	Yes
Truthfulness	Yes	No	No
Time delay	Low	Medium	Low
Time blur	Low	Medium	Low
Possible interfering factors	Vehicle processing capabilities Pedestrian density and background Pedestrian poses Occlusion	Vehicle processing capabilities Pedestrian density Choice of training data Camera sensor quality	Vehicle processing capabilities Pedestrian density and background Pedestrian poses Occlusion Communication overhead

Table 17. Aggregated results

For this use case, all solutions score relatively similar. All solutions offer a high protective effect and score medium on complexity, runtime and implementation effort. Especially a TEE, implemented in each vehicle demonstrates to be a viable solution to effectively anonymize pedestrian data. The possible interfering factors for the model training, such as occlusion or pedestrian density are problems that occur in the model training for all approaches. Factors to focus on in this scenario are processing speed and the protective effect because this might influence the user perception most.

5.4 Use Case: Social Media Location Recommendation

5.4.1 Scenario Description

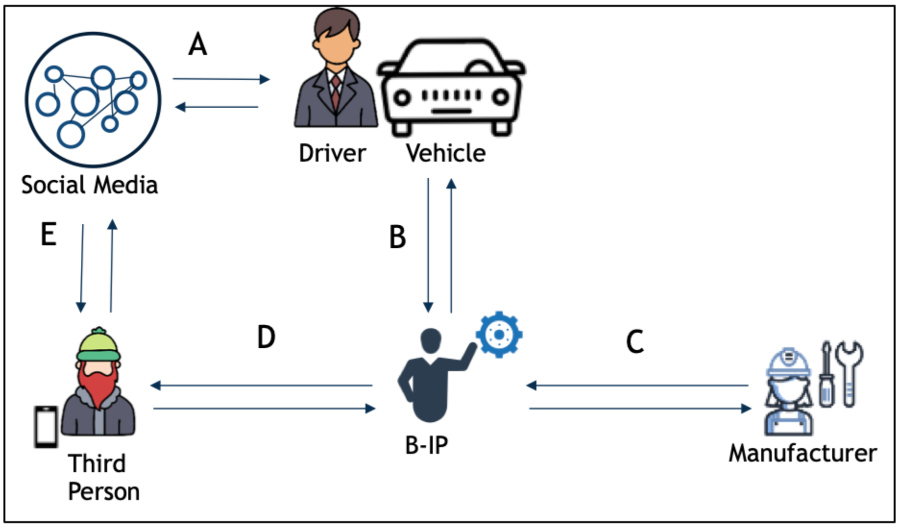


Figure 26 Social Media Services

In-car personalized services are provided by combining internal and external data sources. Car users can access services from social media and platforms in the car. In this example we utilize social media preferences to give recommendations for restaurants. By aggregating information while driving a personalized restaurant recommendation can be made.

Entities:

Driver/Vehicle: Again, in this use-case we treat driver and vehicle as technically one entity. The calculation of a personalized location based on social media preferences is triggered by the driver/vehicle. The driver/vehicle has a communication channel with the social media platform and the B-IP.

Social Media: Information such as recently visited places, food preferences and willingness to pay are stored in a social media platform. The social media platform has a direct communication channel with several users, including the driver and a third person. The social media platform has an indirect communication channel with the B-IP with the driver/vehicle or another device in between. All information that the social media platform might share have to be authorized by the respective user of the social media platform.

B-IP: The B-IP is responsible for calculating location recommendations such as the restaurant recommendation in this use-case. The B-IP receives indirectly information from the social media platform about the users' preferences. Users in this use case are the driver/vehicle and a third person that wants to meet with the driver/vehicle.

Third Person: The third person and the driver/vehicle try to find a restaurant at the intersection of their preferences. Therefore, the third person also authorizes the transmission of social media data between B-IP and the social media platform.

Manufacturer: In this scenario, the manufacturer has no active role. The manufacturer communicates with the B-IP and receives information about model performance and user acceptance.

Steps:

1. Fabian and a third person want to meet at a restaurant that is first, close to both of their locations and second, matches their preferences.
2. Fabian sends a recommendation request to the B-IP with his GPS data and average speed.
3. The B-IP sends a request to share social media data indirectly to the social media platform that has to be authorized by Fabian and the third person. Social media data includes e.g., recently visited places, food preferences or willingness to pay.
4. The social media platform sends the data to the driver/vehicle.
5. The driver/vehicle can alter the data before sending it to the B-IP.
6. The B-IP collects all data and calculates based on the preferences, the location data and an external map with possible locations and a ranking of best matching restaurants.
7. The restaurant with the highest score is sent to Fabian and the third person.
8. The users can evaluate the restaurant and give feedback to the B-IP's recommendation model.

5.4.2 Data Flow Chart

In this section we will go through each communication channel from Figure 26.

Communication channel A: From Vehicle to Social Media

Data	Privacy Sensitivity	Frequency
Authorized request from B-IP	Uncritical	If needed

Communication channel A: From Social Media to Vehicle

Data	Privacy Sensitivity	Data truthfulness at record level	Frequency
Preferences	Critical	Yes	If needed
User ID	Critical	Yes	If needed
Visited places	Critical	Yes	If needed

Communication channel C: From Vehicle to B-IP

Data	Privacy Sensitivity	Data truthfulness at record level	Frequency
GPS	Critical	No	1/s
Speed	Critical	No	1/s
Categorial Preferences	Critical	Yes	1/s
User ID	Uncritical	Yes	1/s
Visited places	Uncritical	Yes	1/s

Communication channel C: From B-IP to Vehicle

Data	Privacy Sensitivity	Data truthfulness at record level	Frequency
Location Recommendation	Critical	Yes	1/s

5.4.3 Attack Scenarios

Location Determination

The B-IP receives the accurate location of a vehicle.

Tracking and Profiling

B-IP collects and stores user preferences, locations and related data and creates user profiles. These profiles can be used by the B-IP for further analysis that are not related to the use case or sold to other parties that are interested in user specific data. Also, a combination with other databases is possible.

Steps:

1. B-IP can easily access preferences from social media and location data.
2. B-IP collects and stores such data over time and creates user profiles.
3. These profiles are then combined with other datasets and sold to other parties interested in such data.

Untrustworthy Recommendations

In this scenario the B-IP returns not the restaurant with the highest score for the users but instead a restaurant that is e.g., paying a fee for being recommended. By this the B-IP creates a competitive advantage through preference for certain restaurants at the expense of user preferences. This scenario has a high probability to cause reputational damage to the manufacturer.

Steps:

1. B-IP makes secret agreements with restaurant operators.
2. B-IP manipulates the used model in favor of the "partner" restaurants
3. B-IP sends manipulated restaurant recommendations to the users

Location Determination and Profiling via Model Inversion attack

A malicious third person can try to reveal the GPS location and preferences of the vehicle/driver. This could happen by attacking the model used by the B-IP to predict the recommendation and GPS location for the joint meeting. In this attack, the training of a machine learning model that simulates the model by the B-IP is necessary. Therefore, the third person requires additional data, such as a map with speed limits and a sample dataset.

Steps:

1. In advance, the third person sends multiple requests with a manipulated vehicle/driver to build an own malicious ML model that makes predictions similar to the model used by the B-IP.
2. The third person receives the joint meeting location and restaurant recommendation for a meeting with a trustworthy vehicle/driver.
3. The third person uses the malicious ML model to back calculate the vehicle/driver's preferences.

5.4.4 Requirements for Approaches to De-identification

- Social media should not automatically receive information about third person
- Persons do not see social media preferences of each other
- B-IP cannot store user specific preferences
- B-IP cannot profile data
- B-IP cannot say with 100% probability whether a restaurant was visited or not
- P-IP cannot manipulate the predictions

5.4.5 Evaluation

Upon accessing the assumption and requirements of the personalized services use-case, all de-identification methods introduced in chapter 4 have been evaluated for their fit for the use case. Only de-identification methods that could initially demonstrate a sufficient level of privacy are discussed in detail below.

Table 18 states methods that have been excluded as well as a brief statement on as to why they are deemed not suitable.

De-identification Technique	**Reason for exclusion**
Sampling	Sample data does not provide privacy protection for the specific sample and relies on a very high sample size which is likely not to be the case. It is not suitable as standalone technique but might be used in addition to others.
Order-preserving encryption	Order of data is not of importance in this use case.
Pseudonymization	Not suitable because profiling is still possible.
Randomization	Randomization is too weak for location perturbation and not suitable for user preferences.
Permutation	Order of data is not of importance in this use case.

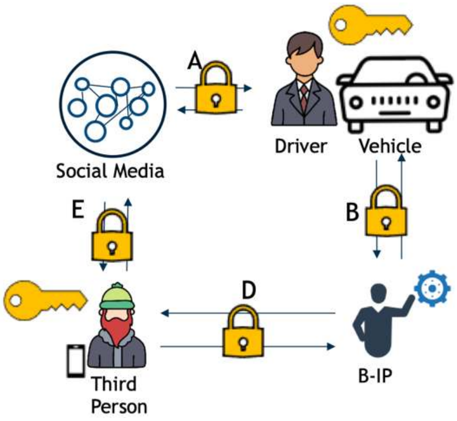
Table 18. Insufficient de-identification techniques for the electricity provider use case

The following de-identification methods are found to be initially suitable and are being discussed in detail in the following:

1. Homomorphic Encryption
2. Secure Multiparty Computation
3. Federated Learning
4. TEE

5.4.6 Approaches to De-identification

Homomorphic Encryption

Diagram	 Social Media A Driver Vehicle E B D Third Person B-IP	
Entities	• *Vehicle/Driver:* The vehicle/driver has already collected the social media data and stored locally. The vehicle/driver encrypts the GPS location and preferences and sends the encrypted data to the B-IP. • *B-IP:* The B-IP receives the encrypted data and calculates the meeting point and restaurant recommendation. The B-IP cannot see the results because they are encrypted. • *Third Person:* Similar to the driver/vehicle the third person encrypts the GPS location and preferences and sends the encrypted data to the B-IP.	
Steps	1. The users (vehicle/driver and third person) encrypt their data (location and preferences). 2. The users send the encrypted data to the B-IP. 3. The B-IP performs the calculation of preferences and a joint meeting location on the encrypted data. 4. The B-IP sends the encrypted results to the users. 5. The users decrypt the results and now know the joint meeting location without knowing the current position of the other person involved.	
Effort	Protective effect	High
	Complexity	High
	Runtime	High
	Degree of maturity	Medium
	Implementation effort	High
	Monetary cost	High
End result data quality	Time delay	Medium
	Time blur	Medium
	Location blur	Medium
	Processing speed	High
	Aggregated data	No
	Truthfulness	Yes
	Time delay	Medium
	Network coverage Processing time	

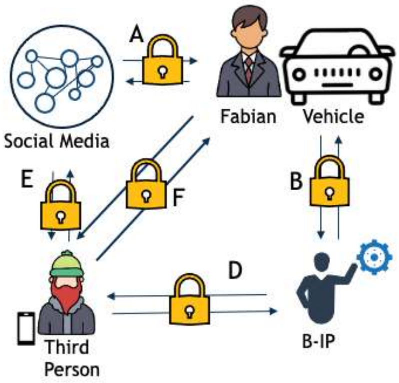
As has been explained in previous chapters the advantage of homomorphic encryption is that data can be computed while it is encrypted, guaranteeing that computations on the data lead to the same results on the decrypted data.

Rohilla et al. [52] provide a prototype, showing how location privacy using homomorphic encryption over the cloud cold be realized. But distinctly from our use case, they assume only one person requesting a location-based service. Bozhon Liu et al. [53], propose the homomorphic encryption scheme and secure indexing (HESI) framework. Their framework assumes two semi trusted servers. One of them is responsible for the logistics and metadata processing, the other performs the computation of the task. In their scenario they assume that requesters want to find a worker in their area without revealing their location.

In our scenario we have two users, the vehicle/driver and the third person who encrypt both and their location data and preferences, and send it in the next step to the B-IP. The preference data of a certain user was already collected from the social media platform. The B-IP who is responsible for the computation can calculate the restaurant that matches both user's preferences on the encrypted data. This could happen similar to the protocols proposed by Rohilla et al. [52].

The presented method highly depends on high computation power in the vehicle for encryption/decryption and at the B-IP to do the calculation in time. Another factor is a good network coverage so that no delay in the communication further slows down the computation of a result.

Secure Multiparty Computation (MPC)

Diagram	 A Fabian Vehicle Social Media E F B D Third Person B-IP
Entities	• *Vehicle/Driver:* The vehicle/driver has already collected the social media data and stored locally. The vehicle/driver together with the third person uses MPC to calculate an average preference score of both preferences. Differential Privacy can be used to perturbate the current GPS position • *B-IP:* The B-IP receives the average preference of driver/vehicle and the third person as well as the locations and speed. The B-IP sends both a location recommendation based of the location and preferences but cannot store specific information about the individual preferences. • *Third Person:* Similar to the driver/vehicle the third person uses MPC to calculate an average preference score from social media data.
Steps	1. MPC can only be used after the vehicle has received the data from the social media provider. 2. The driver/vehicle and the third person use MPC to calculate average preferences. At the same time, the current location is perturbated on the vehicle and the third person's device. 3. The B-IP receives the perturbated locations from vehicle/driver and the third person as well as the average preferences of both. Based on this information the B-IP calculates a restaurant recommendation. 4. The recommendation is sent to the driver/vehicle and the third person.
Effort	Protective effect: High Complexity: High Runtime: High Degree of maturity: High Implementation effort: High Monetary cost: Medium
End result data quality	Time blur: Medium Location blur: Medium Processing speed: High Aggregated data: Yes Truthfulness: No Time delay: Medium
Possible interfering factors	Communication overhead Network coverage

In the MPC the biggest issue is the protocol that can be used to calculate the location recommendation based on geographical and user preference data. In a classic MPC scenario each participant would only see the own input and the generated output of the joint protocol. Although the degree of maturity of MPC in general is “High”, a lot of fine-tuning that increases the implementation costs would be necessary. In practice, there is almost always a tradeoff between efficiency and security causing less secure models, so-called “semi-honest” models [25]. A very similar use case for a recommendation system is provided by Wang et al. [54] who propose a location-aware social point of interest (POI) recommendation system where a recommender (e.g. the service provider) wants to recommend a set of POIs to a certain user (e.g. owner). This is done by calculating a score for POIs that includes the similarities between two users and the distance between a POI and each user’s location. The more far away a location, the lower the score. Finally, they introduce the PLAS protocol that ensures that no sensitive data of the participating parties is disclosed (see Figure 27).

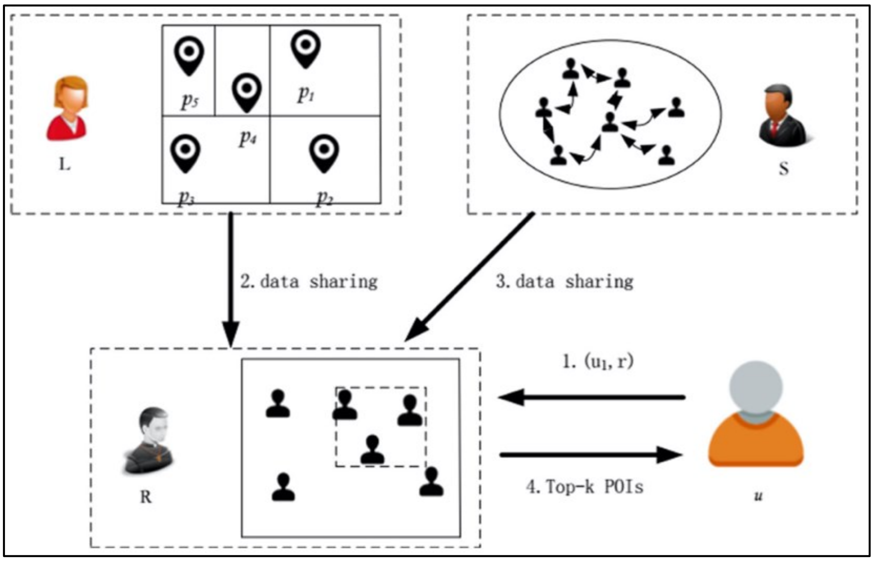
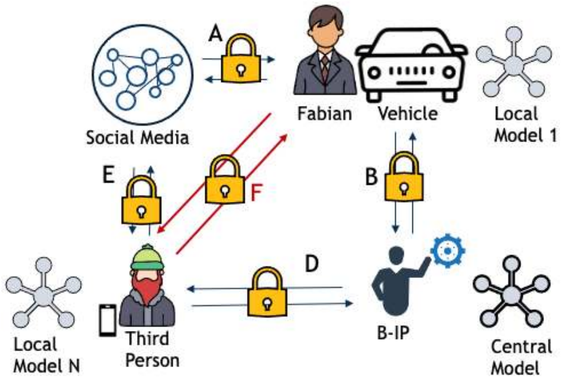


Figure 27 PLAS system model

Possible inferring factors are first, the overhead that is produced by the protocol to calculate the best recommendation between the three parties and second, the network coverage because a stable connection between all entities is required.

Federated Learning with Secure Multiparty Computation

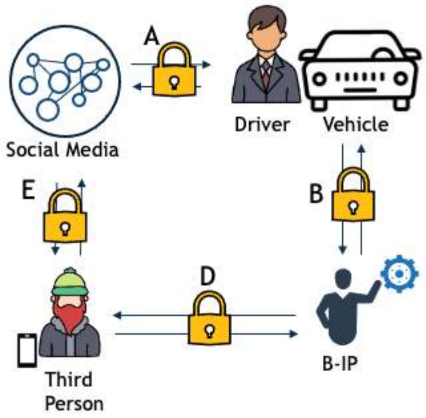
Diagram	 A Social Media Fabian Vehicle Local Model 1 E F B D Local Model N Third Person B-IP Central Model	
Entities	• *Vehicle/Driver:* The vehicle and the third person perform the whole computation locally on the vehicle and the third person's device. No personal data is shared with the B-IP. • *B-IP:* From time to time the B-IP can request gradient updates and send an updated model that then becomes the new local model. • *Third Person:* The vehicle and the third person perform the whole computation locally on the vehicle and the third person's device. No personal data is shared with the B-IP.	
Steps	1. The B-IP trains a central model that contains information about restaurants e.g., price, food, ratings. The model with this information is shared among the users (Vehicle/Driver and third person). 2. The preference data is again requested again from the social media platform and stored on the user's device. 3. With e.g., a secure multiparty computation protocol, the third person and the driver directly exchange location information to calculate the best meeting point with the local data. 4. The users can improve the local model and provide feedback. From time to time the B-IP can request to send gradient updates to improve the central model based on the local models. 5. The new central model is then shared among the users to increase the prediction accuracy.	
Effort	Protective effect	High
	Complexity	High
	Runtime	High
	Degree of maturity	Medium
	Implementation effort	High
	Monetary cost	Medium
End result data quality	Time delay	High
	Time blur	Low
	Location blur	Low
	Processing speed	High
	Aggregated data	No
	Truthfulness	Yes
	Processing speed	

Possible interfering factors	Network coverage

In the federated learning use-case we have chosen a combination of federated learning and MPC. The reason for this is the calculation of the joint meeting place with e.g., GPS data that requires to reveal the location to either another user or the B-IP. To fulfill the requirements of this use case, we come to the result that a combination of federated learning with other technologies might be a good option. While one advantage of federated learning in this scenario is that the critical preference data is stored locally, another advantage is the reduced communication overhead between users and the B-IP. A combination with other protocols such as homomorphic encryption is also possible.

Possible interfering factors are the processing speed on the edge device (user's device) and the network coverage to ensure a fast vehicle to device communication.

Differential Privacy

Diagram	 A Social Media Driver Vehicle E B D Third Person B-IP	
Entities	• *Vehicle/Driver:* The vehicle and the third person use location perturbation to insure with a high probability that their location cannot be disclosed. • *B-IP:* The B-IP is responsible to calculate the location that fits best to the preferences of the third person and the Vehicle/Driver • *Third Person:* The vehicle and the third person use location perturbation to insure with a high probability that their location cannot be disclosed. • *Social Media:* The social media information are sent to the respective person that uses local differential privacy and forwards this information to the B-IP.	
Steps	1. The vehicle/driver sends a computation request to the B-IP. 2. The B-IP sends a request for preference data and location data to the third person and the vehicle/driver 3. The vehicle/driver and the third person collect the respective preference data from the social media platform and use local differential privacy for de-identification. Then, the anonymized data is sent to the B-IP. 4. The B-IP calculates the restaurant with the highest score and sends the result back to the driver/vehicle and the third party.	
Effort	Protective effect	High
	Complexity	High
	Runtime	High
	Degree of maturity	Medium
	Implementation effort	High
	Monetary cost	Medium
End result data quality	Time delay	Medium
	Time blur	Medium
	Location blur	Medium
	Processing speed	Medium
	Aggregated data	Yes
	Truthfulness	Yes
	Processing speed	

Possible interfering factors	Network coverage Communication overhead

In this scenario, differential privacy can on the one hand be used for location perturbation but also on the other hand as local differential privacy for de-identification of the users' preferences. In this scenario we assume that there is no direct communication channel between the driver/vehicle and the third person. The whole communication between the users is done with the B-IP as an intermediary e.g., via an application. After one of the users has send a request to meet with another user of such an application, the B-IP requests the required information (location and preferences) from the users. The users collect this information and use local differential privacy and location perturbation for de-identification. Then the data is shared with the B-IP, who will calculate the best location to meet.

Andrés et al. *[44]* show in their work how location perturbation with differential privacy for location based services could work. The example for restaurant recommendations is explicitly explained in their paper. In comparison to our approach, they focus only on a single user (see Figure 28).

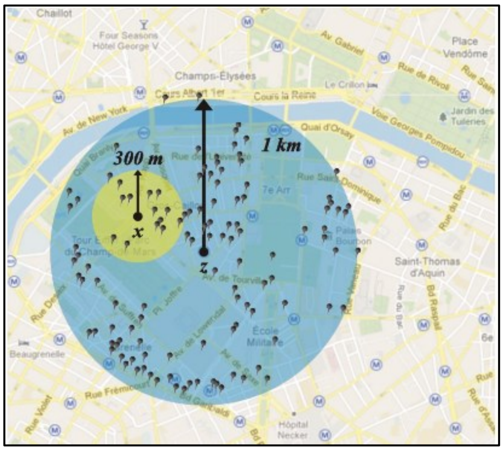


Figure 28: Retrieval information situation for private location based system [44]

One inheriting factors in this scenario is the communication overhead and processing time that can lead to a delay of the service. Although there are several papers about differential privacy, to the best of our knowledge the scenario with a joint meeting location has not been considered so far.

5.4.7 Comparison of Technologies

De-Identification Techniques for use case pedestrian	**Homomorphic Encryption**	**Secure Multiparty Computation (MPC)**	**Federated Learning with Secure Multiparty Computation**	**Differential Privacy**
Protective effect	High	High	High	High
Complexity	High	High	High	High
Runtime	High	High	High	High
Degree of maturity	Medium	High	Medium	Medium
Implementation effort	High	High	High	High
Monetary cost	High	Medium	Medium	Medium
Data Quality				
Time blur	Medium	Medium	Low	Medium
Location blur	Medium	Medium	Low	Medium
Processing speed	Low	High	High	Medium
Aggregated data	No	Yes	No	Yes
Truthfulness	Yes	No	Yes	Yes
Time delay	Medium	Medium	High	Medium
Possible interfering factors	Network coverage Processing time	Communication overhead Network coverage	Processing speed Network coverage	

5.5 Use-Case Predictive Maintenance

5.5.1 Scenario Description

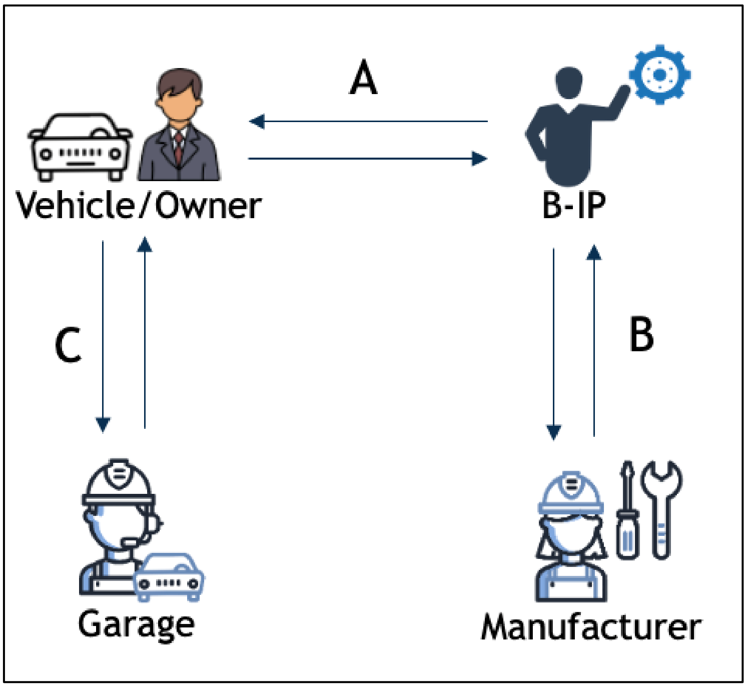


Figure 29: Flow chart of the predictive maintenance use case

The use case of predictive maintenance aims to reduce costs and downtime and optimize service and maintenance intervals by evaluating and analyzing data from a certain car. In our use case we have five entities described as followed: The use case itself deals only lightly with personal data but rather focuses on the possible localization of a vehicle as well as the mapping of driving behavior to a specific vehicle. The use case is included in this work as it was discussed in detail with the project consortium and provides valuable insides on how data can be shared between entities as well as how different types of data can be combined to create new insights from the data that should be protected. Finally, we have defined five entities described as follows:

- *Vehicle/Driver:* The vehicle is equipped with many different sensors that constantly collect and store maintenance related data locally in the vehicle. The vehicle receives a warning from the B-IP if parts are defective or the vehicle needs repair or maintenance. The car can make a repair or maintenance request to the workshop. This must first be approved by the driver. The driver has a communication channel with the garage and can release repair and maintenance orders. Vehicle and driver will be treated as one entity in this scenario, constantly exchanging information.
- *B-IP:* The B-IP takes over the analysis of the vehicle data and has a communication the manufacturer and the vehicle.
- *Manufacturer:* The manufacturer is responsible for providing vehicle model specific information. He also provides information about production defects and recalls. The manufacturer has a communication channel with the B-IP.

- *Garage:* The workshop is responsible for carrying out repairs. Analyses and evaluations that go beyond the actual condition of the vehicle are carried out by the B-IP. The garage receives information from the vehicle about status of the car and the parts that need to be repaired. In case of a defect or a maintenance request, the workshop receives an order from the vehicle.

5.5.2 Data Flow Chart

In this section we will go through each communication channel from Figure 29.

Communication channel A: From vehicle to B-IP

Data	Privacy Sensitivity	Data truthfulness at record level	Frequency
Temperature	uncritical	Yes	1/min
Mechanical noises	uncritical	Yes	1/min
Fluid levels	uncritical	Yes	1/min
Acceleration	medium	Yes	1/min
Altimeter	medium	No	1/min
Speed	medium	No	1/min
Pressure	uncritical	Yes	1/min
Torque	medium	No	1/min
Odometer	uncritical	Yes	1/min
Gasoline consumption	medium	Yes	1/min
ABS	medium	Yes	1/min
VIN	critical	Yes	1/min
Impact angle (at > 45°)	medium	No	1/min
Brake pad	medium	Yes	1/min

Communication channel A: From B-IP to vehicle

Data	Privacy Sensitivity	Frequency
VIN	critical	If needed

Defective parts	medium	If needed
Time urgency	medium	If needed
Report (Justification/ Recommendation)	medium	If needed

Communication channel B: From B-IP to Manufacturer

Data	Privacy Sensitivity	Frequency
General Maintenance Report	uncritical	Quarterly
Car Type	uncritical	Quarterly
Frequency of defective parts	uncritical	Quarterly
Defective parts	uncritical	Quarterly

Communication channel B: From Manufacturer to B-IP

Data	Privacy Sensitivity	Frequency
Request (e.g., Model Update)	Uncritical	If needed

Communication channel D: From Garage to Vehicle/Driver

We only take the predictive maintenance channel into account and not what information the garage can receive when the vehicle is actually in the garage. The repair request might contain critical information e.g., IBAN, name, residence but this information cannot be anonymized.

Data	Privacy Sensitivity	Frequency
Request for repair/maintenance	Critical but necessary	If needed

Communication channel D: From Vehicle/Driver to Garage

Data	Privacy Sensitivity	Frequency
Offer, Bill	Critical but necessary	If needed

Trust assumptions

- *Car:* Trustworthy
- *Driver:* Trustworthy
- *B-IP:* Untrustworthy
- *Manufacturer:* Trustworthy
- *Garage:* Untrustworthy

5.5.3 Attack Scenarios

Motion Profiling

In this attack scenario aims the B-IP to create a motion profile of a specific vehicle that can be identified by the VIN. In combination with a database that contains the mapping of driver and vehicle, a clear motion profile of Fabian can be created.

The motion profiling is realized by observing the sensor data over a long time to identify unique behavior. Examples for this are geographically unique places like mountains that show a unique temperature or height compared to the surroundings. Also, the combination of the impact angle, e.g., at 45° becomes unique after a certain amount of turning operations.

1. Septs B-IP receives and stores sensor data of a certain vehicle.
2. B-IP analyzes the sensor data and combines it with an external database which assigns driver and VIN.
3. The analyzed sensor data reveals the motion profile, e.g., residence, favorite café.
4. In a next step this information could be sold to a third party or used for further attacks.

Driving Behavior Profiling

In this attack scenario the B-IP aims to create a driving behavior profile. Thereby the B-IP clusters users based on the received sensor data. A certain user can then be classified into one of these clusters. Such data could be very interesting e.g., for insurance companies to blacklist users or increase their monthly rate.

Steps:

1. Septs B-IP receives and stores sensor data of a certain vehicle.
2. B-IP analyzes the sensor data a builds clusters with different driving behavior profiles. These clusters can then be evaluated and differentiated into good and bad behavior.
3. Also, a combination with an external database to map further driver data and the VIN is possible.
4. In a next step this information could be sold to a third party or used for further attacks. Interested parties benefit from a risk assessment of a certain user e.g., insurance companies or banks.

Third Party Data Sharing

This scenario is an extension of the above-mentioned scenarios and includes all potentially interested parties the B-IP can share data with. This includes e.g., insurance companies, banks or advertising agencies.

Steps:

1. B-IP receives and stores sensor data of a certain vehicle.
2. The B-IP analyses the data using methods from statistics and machine learning.
3. The B-IP can combine the received data with further datasets to collect more information.
4. The B-IP aims to sell the analyzed data or lists of certain users with a certain behavior, e.g., driving behavior to interested third parties with or without the consent of the user.

Membership Inference Attack

In a membership inference attack a vehicle driver tries to gain information about other participants, the model created by the B-IP was trained on [55]. One option to do this is the transmission of manipulated sensor data to the B-IP.

Steps:

1. Malicious vehicle sends manipulated sensor data
2. B-IP sends back information
3. Malicious vehicle combines received data with an external dataset and can guess with a high probability whether a certain person was part of the training set.
4. The malicious vehicle can disclose the features of a certain user.
5. Based on the disclosed features further attacks such as Motion and Driving Behavior Profiling are possible.

Model Inversion Attack

In a model inversion attack the attacker tries to reveal confident information from the model. In this scenario the garage tries to learn the thresholds of the model used by the B-IP. By manipulating the vehicle during an inspection, the garage can provoke a delayed maintenance message by the B-IP.

Steps:

1. A manipulated vehicle/driver sends multiple malicious requests to the B-IP to build an own malicious ML model that predicts similar to the model used by the B-IP.
2. Once the thresholds are known, vehicles can be manipulated to trigger unnecessary service request.

Upon accessing the assumption and requirements of the personalized services use-case, all de-identification methods introduced in chapter 4 have been evaluated for their fit for the use case. Only de-identification methods that could initially demonstrate a sufficient level of privacy are discussed in detail below. It can be seen that multiple techniques, such as order-preserving encryption, randomization and permutation can be used for the data used in predictive maintenance. However, all these techniques do not provide a sufficient level of privacy and usability on their own. Thus, we focus on the advanced techniques already discussed in the other use cases and include the weaker techniques when suitable. MPC is not usable in this case as aggregated data over multiple vehicles cannot be used to accurately predict failures in a specific vehicle. Differential privacy might be useful to create a predictive maintenance model but cannot be used to process the data of a single vehicle.

5.5.4 Requirements

- Driving patterns and behavior should not be made available to third parties such as insurance companies.
- B-IP and manufacturer should not be able evaluate driving patterns and driving behavior of a vehicle driver.
- The garage should only receive information that is directly related to a repair visit and parts that need to be repaired.

The following de-identification methods are found to be initially suitable and are being discussed in detail in the following:

5.5.5 Evaluation

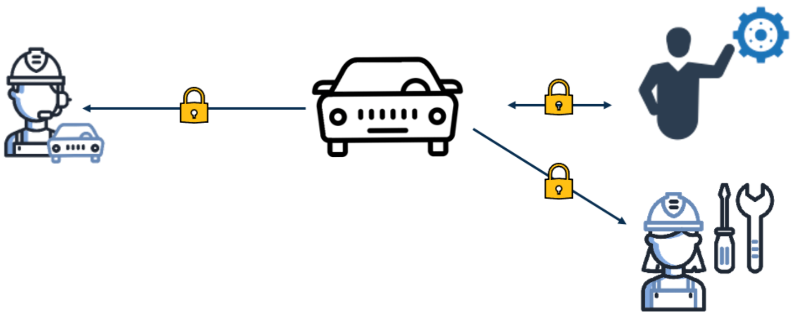
De-identification Technique	**Reason for exclusion**
Sampling	Sampling cannot be used for predictive maintenance because too many information is lost.
Order-preserving encryption	Order of data is not of importance in this use case.
Pseudonymization	Not suitable because likability is necessary.
Randomization	Randomization cannot be used for predictive maintenance because too many information is lost.
Permutation	Order of data is not of importance in this use case.
Differential Privacy	In this scenario differential privacy is only useful to build a predictive maintenance model but not to process the data of a single vehicle.
K-anonymity	Negative impact on model accuracy.

The following de-identification methods are found to be initially suitable and are being discussed in detail in the following:

1. Homomorphic Encryption
2. Federated Learning
3. TEE

5.5.6 Approaches to De-identification

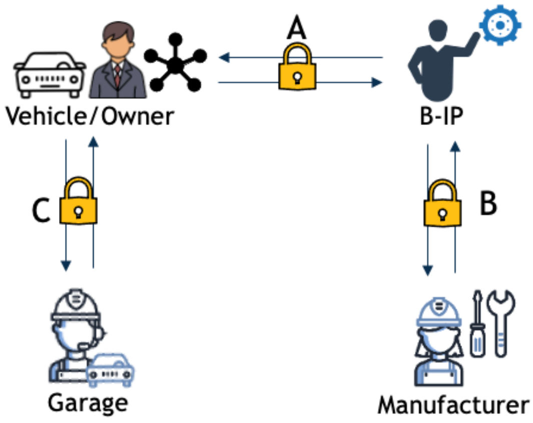
Homomorphic Encryption

Diagram	**Homomorphic Encryption** 	
Entities	• *Vehicle:* Each vehicle collects data on its functioning. Data is k-anonymized in the vehicle. • *Garage:* The garage receives a notification on new parts that need to be ordered for a specific vehicle from that specific vehicle. • *B-IP:* The B-IP receives data from the vehicle and analyzes it. Predicted part failures are communicated to the vehicle. High-level analyses are sent to the vehicle manufacturer in regular time intervals.	
Steps	1. Vehicles collect data while driving on the correct functioning of itself. 2. Vehicle sends data to the B-IP using homomorphic encryption. 3. B-IP analyzes data and sends results back to the vehicle. 4. Vehicle encrypts the data and evaluates whether a garage visit for maintenance is necessary. 5. Vehicle sends maintenance request to the garage as well as information on parts that need to be repaired or ordered. 6. Vehicle sends periodic reports to the vehicle manufacturer.	
Effort	Protective effect	High
	Complexity	High
	Runtime	High
	Degree of maturity	Medium
	Implementation effort	High
	Monetary cost	Medium
End result data quality	Time blur	Medium
	Location blur	Medium
	Processing speed	Low
	Aggregated data	No
	Truthfulness	Yes
	Time delay	High
Possible interfering factors	Vehicle processing capabilities	

Homomorphic encryption can be applied to the predictive maintenance use case with the drawback of an increase in complexity and communication channels. In this scenario, the vehicle homomorphically encrypts the data that is gathered, motion and location information. This data is then sent to the B-IP that analyzes the data. The B-IP is not able to create motion or behavioral profiles of the vehicle driver or owner as the data is encrypted. Similarly, the data cannot be re-identified using external data sources. Similarly, the data is not usable for third parties. The data is then sent back to the vehicle and decrypted.

The scenario now needs to deviate from the initial use case as more tasks need to be taken by the vehicle instead of the B-IP. As the B-IP only computes on decrypted data, the full amount of data needs to be sent to the vehicle instead of an analysis that contains only the results of the predictive maintenance analysis. The vehicle therefore needs to analyze the received data and derive an outcome. This outcome is then shared with the garage in case of needed maintenance. Periodic reports also need to be shared with the vehicle manufacturer whereby deterministic encryption can be applied as the manufacturer is regarded as a trusted entity. Overall, while homomorphic encryption creates a high level of privacy, the technology creates significant overhead at the vehicle. The processing speed is seen as low as time-consuming computations are delegated to the B-IP that then needs to transfer the data back to the vehicle. However, the vehicle itself must act further on the data and needs to communicate with the vehicle manufacturer as well. Different predictive maintenance, e.g., to study degradation of a break or a transmission, would likely need a new setup for the homomorphic encryption as no one-size-fits-all solution exist. Homomorphic encryption is therefore not seen as ideal due to the large runtime, implementation effort and a high level of complexity.

Federated learning

Diagram	**Federated learning**  A Vehicle/Owner B-IP C B Garage Manufacturer	
Entities	• *Vehicle:* Each vehicle collects data on its functioning. The data is directly processed on the vehicle. Gradient updates can be sent to the B-IP. • *Garage:* The garage receives a notification on new parts that need to be ordered for a specific vehicle from that specific vehicle. • *B-IP:* The B-IP trains a central model and distributes this model to the vehicles. The B-IP can request gradient updates from the vehicles and circulate an improved model. • *Vehicle Manufacturer:* The manufacturer receives periodic reports from the B-IP on the condition of the whole fleet.	
Steps	1. Vehicles collects data while driving on the correct functioning of itself. The data is analyzed directly in the local model of the vehicle. 2. The B-IP can request gradient updates to improve the model. 3. The vehicle, through the consent of its driver, can communicate a garage visit to the garage. 4. The garage receives information on planned maintenance and spare parts that need to be ordered.	
Effort	Protective effect	High
	Complexity	High
	Runtime	Medium
	Degree of maturity	Medium
	Implementation effort	High
	Monetary cost	High
End result data quality	Location blur	High
	Processing speed	High
	Aggregated data	No
	Truthfulness	Yes
	Time delay	Low
Possible interfering factors	Communication overhead Untrusted participants Computational power	

Also, for predictive maintenance federated learning has the potential to on the one hand increasing processing speed by computing the results directly in the vehicle and on the other hand improve privacy by not sending sensitive data to an untrusted third party.

One approach for predictive maintenance in IoT edge devices is proposed by Bellavista and Mora [56] in their decentralized learning framework IoTwin (see Figure 30) where they also tested and implemented federated learning.

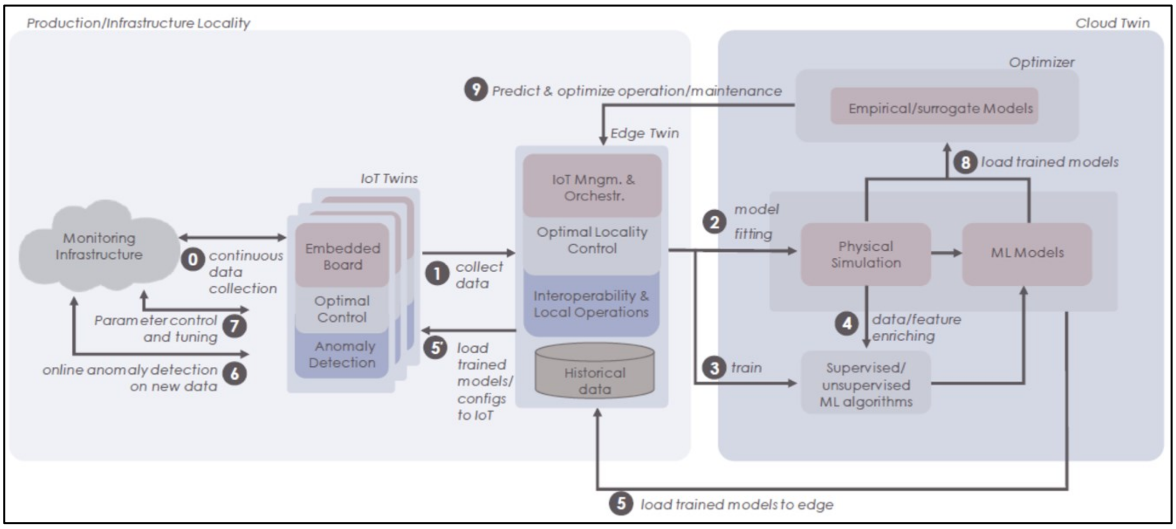
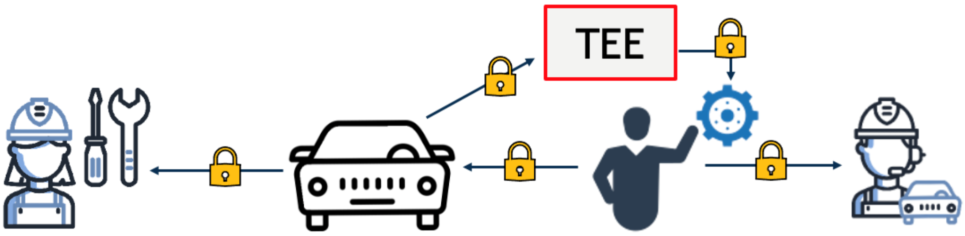

Production/Infrastructure Locality
Cloud Twin
Optimizer
9 Predict & optimize operation/maintenance
Empirical/surrogate Models
Edge Twin
8 load trained models
IoT Twins
IoT Mngm. & Orchestr.
Optimal Locality Control
Interoperability & Local Operations
Historical data
2 model fitting
Physical Simulation
ML Models
Monitoring Infrastructure
0 continuous data collection
Embedded Board
Optimal Control
Anomaly Detection
1 collect data
Parameter control and tuning 7
online anomaly detection on new data 6
5 load trained models/ configs to IoT
4 data/feature enriching
3 train
Supervised/ unsupervised ML algorithms
5 load trained models to edge

Figure 30: IoTwins framework. [56]

In line with the previous federated learning approaches, possible interfering factors for federated learning are the transmission overhead and security issues by malicious participants. Therefore, also in this scenario, a reputation management of devices participating in the federated learning scenario is required. While the communication costs for the model and gradient updates are increased, this communication is not very sensitive to a delayed communication because the central server can start to create a new model when all information are collected. In comparison, the computation time for the result, e.g., a service warning is much more time critical and should not be delayed. Generally, the transmission of data is significantly reduced because a cloud server has not to collect data from all vehicles to update the model or make any prediction. To compute the results locally, more computational power might be required, compared to a cloud-based solution.

Trusted Execution Environment

Diagram	**Trusted Execution Environment**  	
Entities	- *Vehicle:* Each vehicle collects data on its functioning. - *Garage:* The garage receives a notification on new parts that need to be ordered for a specific vehicle from that specific vehicle. - *B-IP:* The B-IP receives data from the vehicle and analyzes it. Predicted part failures are communicated to the vehicle. High-level analyses are sent to the vehicle manufacturer in regular time intervals. - *Vehicle Manufacturer:* The manufacturer receives periodic reports from the B-IP on the condition of the whole fleet.	
Steps	1. Vehicles collect data while driving on the correct functioning of itself. 2. Vehicle sends data to the B-IPs TEE. 3. The data is analyzed in the TEE and its aggregated results readable for the B-IP. 4. The B-IP provides insights from the computed data over multiple vehicles periodically to the vehicle manufacturer. 5. The B-IP provides the vehicle with actionable insights on possible maintenance work. 6. The vehicle, through the consent of its driver, can communicate a garage visit to the garage. 7. The garage receives information on planned maintenance and spare parts that need to be ordered.	
Effort	Protective effect	Medium
	Complexity	High
	Runtime	Medium
	Degree of maturity	Low
	Implementation effort	Medium
	Monetary cost	Medium
End result data quality	Location blur	High
	Processing speed	Medium
	Aggregated data	Yes
	Truthfulness	No
	Time delay	Medium
Possible interfering factors	Security of TEE on B-IP cloud.	

Traditionally, a TEE is employed on mobile devices such as smartphones or tablets. However, recent research demonstrates the use of TEE on cloud servers that may be used for business applications. Intel Software Guard Extensions (SGX) are the TEE that is researched most strongly in that aspect. SGX represents a TEE that is running on later-stage Intel CPUs whereby application can be run in secure containers that are secured using on-chip memory encryption. Access to this memory is mediated by the hardware and only privileged code can add or alter data. Remote parties are able to verify that a specific code is running within an SGX-enclave using a Direct Anonymous Attestation (DAA) scheme [57]. In [57], the authors propose a new solution that "enables dynamic replication and de-commissioning of TEE-based applications in the cloud". The authors find that their solution, named ReplicaTEE, of a cloud TEE remains secure even if an attacker controls a large fraction of the cloud infrastructure. This solution furthermore is found to add only a moderate overhead to existing TEE-based applications. Further work on TEE in a cloud environment exists although the overall research in this area remains limited. In [58], the authors investigate security of user credentials in SGX against Man in the cloud (MITC) attacks in commercial cloud storage solutions by creating a new defense system. The system is found to create only a limited amount of overhead on the client side.

One possible solution for this use case would therefore be to implement a TEE at the B-IPs` cloud server. The vehicle would therefore gather data and transfer it to the B-IP. Within the TEE, agreed upon algorithms would perform predictive maintenance analysis and send encrypted insights back to the vehicle. The vehicle can then contact the garage should the vehicle owner or driver wish to book a repair job. Under the provision that a cloud-based TEE provides a sufficient level of data privacy, this solution offers several benefits. Current research indicates that runtime and overhead are acceptable. Data can be analyzed and aggregated in a cloud environment, preserving the vehicles own resources. The complexity of operations is seen as manageable, although it needs to be noted that cloud-based TEE are still in an early stage. However, several cloud-service providers claim to already offer such services to protect data in use through early-access programs.[6][7][8] No information could be found as to the extent to which the technology is currently leveraged in the existing solutions. Therefore, we argue that this solution is highly complex and not mature as of yet. A viable solution may be expected in the next 2-3 years with a yet unknown degree of protective effect.

[6] See Microsoft Azure Confidential Computing. Available under: https://azure.microsoft.com/en-us/solutions/confidential-compute/#overview (last visited: 28.02.2020)

[7] See IBM Confidential Computing. Available under: https://www.ibm.com/cloud/learn/confidential-computing (last visited: 28.02.2020)

[8] See Porter, Garms and Simakov, 2018. Introducing Asylo: an open-source framework for confidential computing. Available under: https://cloud.google.com/blog/products/identity-security/introducing-asylo-an-open-source-framework-for-confidential-computing (last visited: 28.02.2020)

5.5.7 Comparison of De-identification technologies

De-Identification techniques for use case pedestrian	**Homomorphic Encryption**	**Trusted Execution Environment**	**Federated Learning**
Protective effect	High	Medium	High
Complexity	High	High	High
Runtime	High	Medium	Medium
Degree of maturity	Medium	Low	Medium
Implementation effort	High	Medium	High
Monetary cost	Medium	Medium	High
Data Quality			
Time blur	Medium	High	High
Location blur	Medium	High	High
Processing speed	Low	Medium	High
Aggregated data	No	Yes	No
Truthfulness	Yes	No	Yes
Time delay	High	Medium	Low
Possible interfering factors	Network coverage Processing time	Communication overhead Network coverage	Processing speed Network coverage

6 Outlook

This work presents the current status of academic literature on multiple different de-identification techniques. Our findings demonstrate that there is no single solution that may be equally suitable for every use case in the mobility domain.

In this report, the focus lay on the most-advanced de-identification techniques in order to evaluate their applicability and effectiveness in providing a strong level of privacy for personal data.

It can be seen that the "right" de-identification technique greatly depends on the assumptions and requirements of a use case. If no trusted third party exists, techniques such as MPC and homomorphic encryption provide a high level of privacy at the expense of high computational cost and higher time delay, as compared to other techniques. If data is not to be distributed and analyzed between multiple parties, MPC and federated learning lose their competitive edge. K-anonymity and differential privacy are models that are relatively easy to implement, at the expense of data truthfulness and ultimately a decrease in data usability. A TEE might provide a high level of privacy and could be implemented within vehicles, and even in cloud solutions of business analytics providers. However, the correct implementation and execution of a TEE still requires some degree of trust. The application of cloud-based TEE is still in its early stages, more research in academia is needed. Oftentimes, the combination of different techniques, such as differential privacy and federated learning, lead to solutions that offer a stronger privacy guarantee than its individual components, at the expense of increased complexity. The literature review demonstrated that academia is currently focusing on homomorphic encryption and MPC whereas research in TEE and confidential computing is somewhat limited. Indeed, more research on TEE could be of great value for vehicle manufacturers as mobility-related use cases would greatly benefit from TEE that could be implemented at third party cloud providers or directly in a vehicle. Advancements in federated learning and MPC would especially benefit vehicle fleet providers and use cases in which multiple vehicles gather and share data collaboratively. Overall, homomorphic encryption generally provides the highest level of privacy and depicts a solution that could be used in nearly every use cases. The drawbacks of this technique, runtime and a highly limited number of possible operations, remain however. These problems are continuously addressed in academia and specialized solutions are being created as proof-of-concepts. Further research that combines traffic simulation software, as depicted in chapter 4.1, with real vehicle data might be utilized to evaluate the effectiveness of the solutions outlined in chapters 4.2 – 4.5. Here, more information on the specific types and frequency of data that is being used in vehicles needs to be provided to generate results that provide measurable insights.

Ultimately, the decision for one de-identification technique will not only rely on technical considerations but also on a legal evaluation of these techniques. Similar to the technical evaluation, a legal assessment is likely to consider both, the techniques in general as well as the specifics of their implementation in a particular scenario.

7 References

1. Dwork C, McSherry F, Nissim K, Smith A (2006) Calibrating noise to sensitivity in private data analysis. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)

2. Nelson B, Olovsson T (2018) Introducing differential privacy to the automotive domain: Opportunities and challenges. IEEE Veh Technol Conf 2017-Septe:1–7. https://doi.org/10.1109/VTCFall.2017.8288389

3. Tschantz MC, Sen S, Datta A (2020) SoK: Differential privacy as a causal property. Proc - IEEE Symp Secur Priv 2020-May:354–371. https://doi.org/10.1109/SP40000.2020.00012

4. Li X, Zhang H, Ren Y, et al (2020) PAPU: Pseudonym Swap with Provable Unlinkability Based on Differential Privacy in VANETs. IEEE Internet Things J 1–1. https://doi.org/10.1109/jiot.2020.3001381

5. Ghane S, Jolfaei A, Kulik L, et al (2020) Preserving Privacy in the Internet of Connected Vehicles. IEEE Trans Intell Transp Syst 1–10. https://doi.org/10.1109/tits.2020.2964410

6. Yang W, Sun YE, Huang H, et al (2019) Persistent transportation traffic volume estimation with differential privacy. Proc - 2019 IEEE SmartWorld, Ubiquitous Intell Comput Adv Trust Comput Scalable Comput Commun Internet People Smart City Innov SmartWorld/UIC/ATC/SCALCOM/IOP/SCI 2019 566–573. https://doi.org/10.1109/SmartWorld-UIC-ATC-SCALCOM-IOP-SCI.2019.00136

7. Liu B, Xie S, Wang H, et al (2019) VTDP: Privately Sanitizing Fine-grained Vehicle Trajectory Data with Boosted Utility. IEEE Trans Dependable Secur Comput PP:1. https://doi.org/10.1109/TDSC.2019.2960336

8. Ma Z, Zhang T, Liu X, et al (2019) Real-Time Privacy-Preserving Data Release. 68:8091–8102

9. Gentry C (2009) Fully Homomorphic Encryption Using Ideal Lattices. In: Proceedings of the Annual ACM Symposium on Theory of Computing

10. Sun C, Liu J, Jie Y, et al (2018) Ridra: A rigorous decentralized randomized authentication in VANETs. IEEE Access 6:50358–50371. https://doi.org/10.1109/ACCESS.2018.2868417

11. Zhang J, Yang F, Ma Z, et al (2020) A Decentralized Location Privacy-Preserving Spatial Crowdsourcing for Internet of Vehicles. IEEE Trans Intell Transp Syst 1–15. https://doi.org/10.1109/tits.2020.3010288

12. Farouk F, Alkady Y, Rizk R (2020) Efficient Privacy-Preserving Scheme for Location Based Services in VANET System. IEEE Access 8:60101–60116. https://doi.org/10.1109/ACCESS.2020.2982636

13. Kong Q, Lu R, Ma M, Bao H (2019) A privacy-preserving sensory data sharing scheme in Internet of Vehicles. Futur Gener Comput Syst 92:644–655.

https://doi.org/10.1016/j.future.2017.12.003

14. Raja G, Anbalagan S, Vijayaraghavan G, et al (2020) Energy-Efficient End-to-End Security for Software Defined Vehicular Networks. IEEE Trans Ind Informatics 3203:1–1. https://doi.org/10.1109/tii.2020.3012166

15. Samarati P, Sweeney L (1998) Protecting Privacy when Disclosing Information: k-Anonymity and its Enforcement Through Generalization and Suppresion. Proc IEEE Symp Res Secur Priv

16. Wang J, Cai Z, Yu J (2020) Achieving Personalized k-Anonymity-Based Content Privacy for Autonomous Vehicles in CPS. IEEE Trans Ind Informatics 16:4242–4251. https://doi.org/10.1109/TII.2019.2950057

17. Corser GP, Fu H, Banihani A (2016) Evaluating Location Privacy in Vehicular Communications and Applications. IEEE Trans Intell Transp Syst 17:2658–2667. https://doi.org/10.1109/TITS.2015.2506579

18. Falamas DE, Marton K (2019) Performance Impact Analysis of Rounds and Amounts of Communication in Secure Multiparty Computation Based on Secret Sharing. Proc - RoEduNet IEEE Int Conf 2019-Octob:1–6. https://doi.org/10.1109/ROEDUNET.2019.8909467

19. Von Maltitz M, Bitzer D, Carle G (2019) Data querying and access control for secure multiparty computation. 2019 IFIP/IEEE Symp Integr Netw Serv Manag IM 2019 171–179

20. Hastings M, Hemenway B, Noble D, Zdancewic S (2019) SoK: General purpose compilers for secure multi-party computation. Proc - IEEE Symp Secur Priv 2019-May:1220–1237. https://doi.org/10.1109/SP.2019.00028

21. Ghanem SM, Moursy IA (2019) Secure Multiparty Computation via Homomorphic Encryption Library. Proc - 2019 IEEE 9th Int Conf Intell Comput Inf Syst ICICIS 2019 227–232. https://doi.org/10.1109/ICICIS46948.2019.9014698

22. Sayyad S (2020) Privacy Preserving Deep Learning using Secure Multiparty Computation. Proc 2nd Int Conf Inven Res Comput Appl ICIRCA 2020 139–142. https://doi.org/10.1109/ICIRCA48905.2020.9183133

23. Li M, Andersen DG, Park JW, et al (2014) Scaling distributed machine learning with the parameter server. In: Proceedings of the 11th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2014

24. Konečný J, McMahan HB, Yu FX, et al (2016) Federated Learning: Strategies for Improving Communication Efficiency. Iclr

25. Yang Q, Liu Y, Chen T, Tong Y (2019) Federated machine learning: Concept and applications. ACM Trans Intell Syst Technol. https://doi.org/10.1145/3298981

26. Li T, Sahu AK, Talwalkar A, Smith V (2020) Federated Learning: Challenges, Methods, and Future Directions. IEEE Signal Process Mag. https://doi.org/10.1109/MSP.2020.2975749

27. Liu Y, Yu JJQ, Kang J, et al (2020) Privacy-Preserving Traffic Flow Prediction: A

Federated Learning Approach. IEEE Internet Things J. https://doi.org/10.1109/JIOT.2020.2991401

28. Xu C, Mao Y (2020) An improved traffic congestion monitoring system based on federated learning. Inf. https://doi.org/10.3390/INFO11070365

29. Khan MA, Kulkarni P, El Sayed H (2020) Inter-stakeholders Relationship in the Envisioned Autonomous Driving Era. In: ACM International Conference Proceeding Series

30. Kaissis GA, Makowski MR, Rückert D, Braren RF (2020) Secure, privacy-preserving and federated machine learning in medical imaging. Nat Mach Intell. https://doi.org/10.1038/s42256-020-0186-1

31. Schwendicke F, Samek W, Krois J (2020) Artificial Intelligence in Dentistry: Chances and Challenges. J Dent Res. https://doi.org/10.1177/0022034520915714

32. Czeizler E, Wiessler W, Koester T, et al (2020) Using federated data sources and Varian Learning Portal framework to train a neural network model for automatic organ segmentation. Phys Medica. https://doi.org/10.1016/j.ejmp.2020.03.011

33. Bogdanova A, Attoh-Okine N, Sakurai T (2020) Risk and Advantages of Federated Learning for Health Care Data Collaboration. ASCE-ASME J Risk Uncertain Eng Syst Part A Civ Eng. https://doi.org/10.1061/ajrua6.0001078

34. Kawa D, Punyani S, Nayak P, et al (2019) Credit Risk Assessment from Combined Bank Records using Federated Learning. Int Res J Eng Technol 1355

35. Wang G, Dang CX, Zhou Z (2019) Measure Contribution of Participants in Federated Learning. In: Proceedings - 2019 IEEE International Conference on Big Data, Big Data 2019

36. Sabt M, Achemlal M, Bouabdallah A (2015) Trusted execution environment: What it is, and what it is not. In: Proceedings - 14th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2015

37. Liu L, Li J, Yuan TT (2020) TEE-based mutual proofs of transmission services in decentralized systems. IEEE INFOCOM 2020 - IEEE Conf Comput Commun Work INFOCOM WKSHPS 2020 754–759. https://doi.org/10.1109/INFOCOMWKSHPS50562.2020.9162764

38. Vinayagamurthy D, Gribov A, Gorbunov S (2017) StealthDB: A scalable encrypted database with full SQL query support. arXiv 2019:370–388. https://doi.org/10.2478/popets-2019-0052

39. Wagh S, Cuff P, Mittal P (2018) Differentially Private Oblivious RAM. Proc Priv Enhancing Technol 2018:64–84. https://doi.org/10.1515/popets-2018-0032

40. Mubashwir Alam AKM, Sharma S, Chen K (2020) SGX-MR: Regulating dataflows for protecting access patterns of Data-Intensive SGX Applications. arXiv 2021:5–20. https://doi.org/10.2478/popets-2021-0002

41. Cerdeira D, Santos N, Fonseca P, Pinto S (2020) SoK: Understanding the Prevailing Security Vulnerabilities in TrustZone-assisted TEE Systems. Proc - IEEE Symp Secur

Priv 2020-May:1416–1432. https://doi.org/10.1109/SP40000.2020.00061

42. Li T, Lin L, Gong S (2019) AutoMPC: Efficient multi-party computation for secure and privacy-preserving cooperative control of connected autonomous vehicles. In: CEUR Workshop Proceedings

43. Bittau A, Erlingsson Ú, Maniatis P, et al (2017) PROCHLO: Strong Privacy for Analytics in the Crowd. In: SOSP 2017 - Proceedings of the 26th ACM Symposium on Operating Systems Principles

44. Andrés ME, Bordenabe NE, Chatzikokolakis K, Palamidessi C (2013) Geo-indistinguishability: Differential privacy for location-based systems. In: Proceedings of the ACM Conference on Computer and Communications Security

45. Saputra YM, Hoang DiT, Nguyen DiN, et al (2019) Energy demand prediction with federated learning for electric vehicle networks. In: 2019 IEEE Global Communications Conference, GLOBECOM 2019 - Proceedings

46. Yin F, Lin Z, Xu Y, et al (2020) FEDLOC: Federated learning framework for data-driven cooperative localization and location data processing. arXiv

47. Cheu A, Smith A, Ullman J, et al (2019) Distributed differential privacy via shuffling. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)

48. Hukkelås H, Mester R, Lindseth F (2019) DeepPrivacy: A Generative Adversarial Network for Face Anonymization. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics) 11844 LNCS:565–578. https://doi.org/10.1007/978-3-030-33720-9_44

49. Chamikara MAP, Bertok P, Khalil I, et al (2020) Privacy Preserving Face Recognition Utilizing Differential Privacy. Comput Secur 97:. https://doi.org/10.1016/j.cose.2020.101951

50. Luo J, Wu X, Luo Y, et al (2019) Real-world image datasets for federated learning. arXiv

51. Elbir AM, Coleri S (2020) Federated Learning for Vehicular Networks. arXiv

52. Rohilla A, Khurana M, Singh L (2017) Location Privacy using Homomorphic Encryption over Cloud. I J Comput Netw Inf Secur 8:32–40. https://doi.org/10.5815/ijcnis.2017.08.05

53. Liu B, Chen L, Zhu X, et al (2017) Protecting location privacy in spatial crowdsourcing using encrypted data. In: Advances in Database Technology - EDBT

54. Wang W, Liu A, Li Z, et al (2019) Protecting multi-party privacy in location-aware social point-of-interest recommendation. World Wide Web. https://doi.org/10.1007/s11280-018-0550-9

55. Shokri R, Stronati M, Song C, Shmatikov V (2017) Membership Inference Attacks Against Machine Learning Models. In: Proceedings - IEEE Symposium on Security and Privacy

56. Bellavista P, Mora A (2019) Edge cloud as an enabler for distributed AI in industrial IoT applications: The experience of the iotwins project. In: CEUR Workshop Proceedings

57. Soriente C, Karame G, Li W, Fedorov S (2019) ReplicaTEE: Enabling seamless replication of SGX enclaves in the cloud. In: Proceedings - 4th IEEE European Symposium on Security and Privacy, EURO S and P 2019

58. Liang X, Shetty S, Zhang L, et al (2017) Man in the Cloud (MITC) Defender: SGX-Based User Credential Protection for Synchronization Applications in Cloud Computing Platform. In: IEEE International Conference on Cloud Computing, CLOUD