Lessons from a Robotaxi: Challenges in Selecting Privacy-Enhancing Technologies

Ala'a Al-Momani[1][0000−0001−5752−7338], David Balenson[2][0000−0002−0913−4852], Christoph Bösch[3][0000−0001−9312−8000], Zoltán Ádám Mann[4][0000−0001−5741−2709], Sebastian Pape[5,6][0000−0002−0893−7856], and Jonathan Petit[7][0000−0002−8644−1442]

[1] Ulm University
[2] USC Information Sciences Institute
[3] Bosch Research
[4] University of Halle-Wittenberg
[5] Continental Automotive Technologies GmbH
[6] Goethe University Frankfurt
[7] Qualcomm Technologies Inc.

Abstract. Engineering privacy-friendly systems requires first assessing privacy threats and then selecting privacy-enhancing technologies (PETs) to mitigate the threats. While well-established methods such as LINDDUN support threat assessment, systematic approaches for PET selection remain underdeveloped. This paper presents our experience applying three such approaches to a realistic robotaxi use case. Although each method has been validated by its respective authors on simple use cases, we found that none could adequately support PET selection in our complex, real-world scenario. As a result, we also explored a pragmatic approach based on Hoepman's privacy strategies. By analyzing the strengths and limitations of these approaches, we identify key challenges that PET selection methodologies should address and provide recommendations to guide the future development of such methodologies.

Keywords: privacy-enhancing technologies · PET selection · privacy threats · privacy threat mitigation · privacy engineering · robotaxi.

1 Introduction

For the early phases of the privacy engineering process — such as privacy threat assessment — several methodologies provide specific guidance (e.g., LINDDUN [28], PANOPTIC [18], and xCOMPASS [9]). These methodologies support the high-level design of privacy-friendly systems reasonably well, often through the use of privacy strategies and privacy patterns [13]. Academic efforts have also proposed ways to support later phases, in particular the selection of Privacy-Enhancing Technologies (PETs) to address the found privacy threats. Such work draws on privacy principles [24], best practices, activities, objectives, patterns [17, 25], strategies [13], and threat models [8], as well as the broader concept of privacy

by design [11]. However, the practical applicability of these proposals is not fully understood. Applying them to the detailed design of privacy-friendly systems in the real world may be challenging because of the approaches' high level of abstraction and other limitations and shortcomings.

This work investigates how the PET selection problem can be solved in practice, using a realistic robotaxi system as use case. Robotaxi services involve extensive and sensitive data processing throughout their lifecycle — from ride requests and routing to post-ride analytics — making them an ideal testbed for evaluating PET selection methodologies. Our aim is to investigate to what extent existing methodologies can be used to select appropriate PETs to enhance the privacy in the considered robotaxi service. In this work, we do not propose the final design of a privacy-preserving robotaxi service, but rather focus on investigating the methodologies for selecting PETs.

We make the following contributions: i) We identify three methodologies in the literature that promise guidance on PET selection, and apply them to a realistic robotaxi use case. We find that none yield satisfactory results. ii) We apply a pragmatic, experience-based approach based on Hoepman's privacy strategies [13] to identify a useful set of PETs. iii) We analyze the strengths and limitations of these approaches and extract insights to inform the development of improved PET selection methodologies. Our findings show that existing methodologies provide limited — or no — support for the detailed design and actual implementation of privacy-friendly systems. In particular, there is a lack of systematic, actionable support for selecting PETs as well as clear guidance how to implement and configure the selected PETs, how to combine them effectively, and how to integrate them into an overall system.

2 Related Work

We identified several privacy frameworks and projects. They cover the areas of privacy engineering (STRAP [15], which builds on prior work by Bellotti and Sellen [6] and Hong et al. [14]), system re-engineering (POSD [5]), privacy by design (PRIPARE[8] based on the work of Kung [19] and Hoepman [13]), and compliance (PARROT [4]). MITRE has released the Privacy Engineering Framework and Life Cycle Adaptation Guide[9], while ENISA has published the PETs Control Matrix[10] and a report on data protection engineering[11]. However, none of these frameworks give specific support in the selection of PETs.

Several relevant standards also exist. ISO/IEC 27701 extends ISO/IEC 27001 by adding requirements for establishing and improving a Privacy Information Management System (PIMS). ISO/IEC 27550 describes privacy engineering across the system lifecycle, drawing from Hoepman's privacy strategies [13] and Privacy

[8] https://pripareproject.eu/

[9] https://www.mitre.org/sites/default/files/2021-11/

[10] https://www.enisa.europa.eu/news/enisa-news/enisas-pets-control-matrix-a-tool-to-evaluate-online-and-mobile-privacy-tools

[11] https://www.enisa.europa.eu/publications/

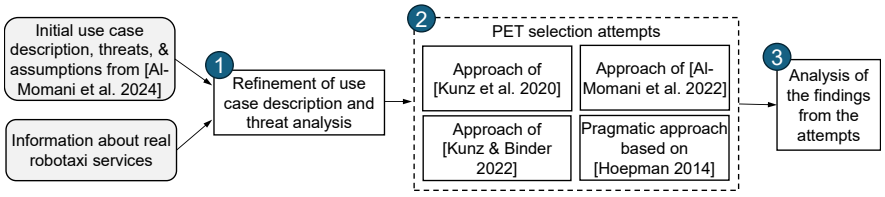

Fig. 1: Overview of the methodology used in this paper

Control Examples that are similar to patterns (e.g., Hide: Encryption, Mixing, Perturbation). Similar to NIST SP800-53, ISO/IEC 29151 defines objectives, controls, and guidelines for implementing controls for protecting personally identifiable information (PII). Yet, none of these standards provide specific support for selecting PETs.

In the academic literature, Drozd and Dürmuth [10] suggested linking privacy patterns to PETs, but only as a conceptual outlook. Pape et al. [24] proposed selecting PETs based on GDPR principles, without referencing specific threats. Adams [1] introduced a privacy tree to classify PETs, offering some guidance for selection, but the list is incomplete and several leaves are linked to multiple PETs. Jordan et al. [16] provide an extensive list of PETs, but offer minimal support for selecting. We only found three papers that provide specific guidance in PET selection [3, 20, 21], which we discuss in greater detail in Section 5.

As our use case is in the automotive domain, we also examined PET-related literature in this area. Al-Momani et al. [2] explored the usefulness of privacy patterns in improving privacy in future automotive systems. Chah et al. [7] applied LINDDUN to analyze privacy threats. Pape et al. [26] proposed a system model to identify suitable integration points for PETs in a vehicle. Löbner et al. [22] evaluated de-identification techniques in automotive use cases. None of these works proposed a methodology for selecting suitable PETs.

3 Methodology

Fig. 1 depicts the methodology used to perform the research reported in this paper. Our methodology is structured around a *refined robotaxi use case* derived from Al-Momani et al. [2]. We enhanced this use case to reflect more realistic data flows and service phases based on descriptions from real providers like Waymo and Uber[12]. We carefully checked that these refinements did not alter the original threat model or its underlying assumptions. As a result, we were able to reuse the *threat assessment* conducted by Al-Momani et al.[2].

To *identify suitable PETs* for our use case, we applied three PET selection approaches from the literature: i) Kunz et al. [20] who propose a reproducible method for selecting data-dependent PETs that can be used independently or

[12] cf. `https://waymo.com` and `https://www.uber.com`, respectively

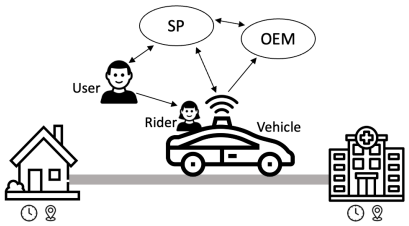

Fig. 2: Basic system model of a robotaxi service, from [2].

alongside other methods; ii) Kunz and Binder [21] who offer an application-oriented classification of PETs based on privacy protection goals, functional context, technology maturity, and impact on various non-functional requirements; and iii) Al-Momani et al. [3] who employ decision trees to guide the selection of privacy solutions based on LINDDUN threats and Hoepman's privacy strategies [13]. In addition to these approaches, we applied a *pragmatic, experience-driven approach* (cf. Sect. 5.4) in which we revisited assumptions, analyzed the purpose of data processing, and considered applicable PETs. We then *analyzed the outcomes* to uncover key challenges, limitations, and differences across the approaches. All steps and findings were collaboratively reviewed to ensure consistency.

4 Use Case: Robotaxi – Refined System Model

Robotaxi services, which are autonomous, driverless taxi systems, represent a cutting-edge application of self-driving vehicle technology. By focusing on a generic robotaxi service, our aim is to derive insights applicable across the broader industry, rather than to a single provider. From a privacy perspective, a robotaxi service differs significantly from a traditional taxi service. In a traditional taxi, the driver handles not only the driving, but also rider interaction, payment, and unexpected situations. In a robotaxi, these functions are performed by a combination of artificial intelligence and a remote service provider. As a result, more data may need to be collected to ensure safe and effective service operation.

Our system model builds on the robotaxi model proposed by Al-Momani et al. [2], providing a refined system version that offers closer alignment with real-world deployments. This refinement is based on examining existing services and incorporates best practices from the industry. While it does not (intentionally) address privacy enhancements, the refined model serves as a more practical foundation for selecting applicable PETs to mitigate the identified privacy threats.

Additionally, we noticed during the application of the pragmatic approach that all of the three investigated approaches require a clean use case description with minimal assumptions. Therefore, we revisited the original assumptions, asking if the data in question was truly necessary and if it could be reduced. For instance, we challenged the assumption that a user's birth date needs to be collected during registration, as a more privacy-friendly option would be to use

Table 1: Data collected or assigned and data used in the various phases.

Phase		1								2							3				4					5				6			
Action		Account creation								Booking							V. assign.				Ride					Payment				Post-ride			
Data collected or assigned		user_id	Name	Address	e-mail	Phone	Payment	legal age	(Preferences)	PLoc(s)	PTime	DLoc(s)	Route	No. of riders	Rider names		Vehicle id	ETA	User location	Fleet manag. data	Vehicle Location	Camera (internal)	Audio (internal)	Sensors (internal)	User interaction	Fare	Payment method	Ride history	User feedback	Feedback	Incident details	Social sharing	...
Data used in:	P1	✓	✓	✓	✓	✓	✓	✓	✓																							
	P2	✓	✓	.	.	.	.	.	✓	✓	✓	✓	✓	✓	✓																	
	P3	✓	.	.	✓	✓	.	.	✓	✓	✓	✓	✓	✓	.		✓	✓	✓	✓												
	P4	✓	.	.	.	.	.	.	✓	✓	✓	✓	✓	.	.		.	.	✓	✓	✓	✓	✓	✓	✓								
	P5	✓	✓	✓	.	.	✓	.	?	✓	✓	✓	✓	✓	.		.	.	.	.	.	.	.	.	.	✓	✓	✓	✓				
	P6	✓	?	?	?	?	?	.	?	?	?	?	?	?	?		?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	.

Pn: Phase n ✓: data needed ?: data potentially needed, depending on post-ride actions.

just a binary check (e.g., "user is of legal age") which avoids the collection of the full date of birth, which could be used for identification.

As shown in Fig. 2, the robotaxi system involves four primary parties: *User*, who requests and manages rides through an application; *Rider*, the individual taking the ride, who may or may not be the same as the User; Service Provider, *SP*, who operates the robotaxi service, manages the backend system, assigns vehicles, and ensures smooth operation; and Original Equipment Manufacturer, *OEM*, who builds and maintains the vehicle, including hardware and software updates. In addition to these natural persons (*User* and *Rider*) and legal entities (*SP* and *OEM*), *Vehicle* can be seen as a fifth party.

The use of a robotaxi service involves several phases, each requiring specific data elements for effective operation. In the following, we describe each phase. Table 1 summarizes the data collected or assigned during these phases, along with the specific phases in which each data item is used or required.

1. Account Creation. *User*s create an account through an application. **Data Collected:** Personal information such as name, email address, phone number, and payment details (e.g., credit card information). **Purpose:** To authenticate users, enable payment processing, and establish a user profile for service access. **Additional Features:** *User*s may also indicate preferences such as accessibility needs (e.g., wheelchair-accessible vehicles), select other service-specific options, or participate in a loyalty program.

2. Booking a Ride. *User*s input their desired pickup and drop-off location(s) into the app, and optionally specify a pick-up time, number of riders, and specific preferences (e.g. vehicle features). **Data Collected:** Current location (via GNSS), pickup location, drop-off location, and potentially pick-up time and preferred routes. **Purpose:** To generate ride requests and facilitate assignment of a vehicle to *User* in the next phase. **Additional Features:** *User*s receive confirmation notifications, and the app provides options to adjust the booking if needed. If the taxi is booked for a different *Rider*, the name is provided by *User*.

3. Vehicle Assignment & Ride Confirmation. The system assigns an autonomous vehicle and provides ride details to *User*. **Data Collected:** Vehicle identification (e.g., make, model, license plate), estimated time of arrival (ETA), and *Rider*'s updated location for precise pickup (if selected). **Purpose:** To inform *User* of vehicle details and ensure accurate pickup coordination. **Additional Features:** *User* is notified when the vehicle arrives. Identity confirmation (e.g., PIN) is required to ensure the correct *Rider* enters the vehicle. Additionally, the vehicle assignment requires fleet management data, including the precise location of vehicles and the current fuel or battery levels.

4. Ride Execution. The autonomous vehicle navigates to the destination, guided by its sensors and real-time data processing. **Data Collected:** Real-time vehicle location, internal and external sensor data (e.g., audio, cameras, LIDAR) and user interaction data within the vehicle (e.g., temperature or music preferences). Sensor data, camera data, and vehicle location are also accessible to the *OEM* at any time. **Purpose:** To enable safe travel, ensure *Rider* comfort, and provide operational support. **Additional Features:** *Rider* may change the route or drop-off location and can contact customer support via vehicle interface or the app if issues arise.

5. Payment and Feedback. Payment is processed automatically upon ride completion. *Rider* can provide feedback via the vehicle interface, and *User* via the application. **Data Collected:** Ride fare details, payment method, trip history, and user feedback (e.g., ratings, comments). **Purpose:** To complete the financial transaction, maintain a record of rides, and improve service quality based on feedback. **Additional Features:** *User* may receive trip summaries, and promotional offers or discounts are applied based on *User*'s profile.

6. Post-Ride Actions. Additional interactions may occur between *User* and *SP*, including invoice creation, ride history and analytics, customer support, loyalty programs and rewards, safety and security issues, service customization, data deletion, subscription cancellation, and social media sharing. **Data Use:** Depending on the action, different existing data items may be reused or new data may be collected.

5 PET Selection

Al-Momani et al. [2] conducted a privacy threat assessment of the original use case. Because our refined use case closely aligns with the original, particularly in terms of privacy threats, the assessment remains applicable, and we refer readers to the original paper for more details. Our current focus is on selecting PETs to mitigate these threats.

Our literature review identified three approaches that offer specific guidance for PET selection. In Sections 5.1-5.3, we describe our experience applying these methods to the robotaxi use case. Given the limitations we encountered, we also applied a pragmatic approach based on Hoepman's privacy strategies [13]. The challenges reported in Sections 5.1-5.4 are not intended as criticisms of these approaches. We recognize these approaches are valuable initial steps toward

addressing a complex problem. Our goal is to highlight that the current state of the art in PET selection remains inadequate for handling realistic use cases.

5.1 Approach of Kunz et al. (2020)

Kunz et al. [20] proposed a methodology for selecting PETs for IoT-based services, with a focus on the automotive domain. The methodology consists of four steps: service description, data-driven elicitation, service-driven elicitation, and PET selection. We go through these four steps and try to apply them to our use case.

A. Service description. In this step, the service is specified, focusing on the required data and the purposes of data processing. We have done this in Sect. 4.

B. Data-driven elicitation. In this step, all data identified in the first step is analyzed according to 6 criteria: continuous or categorical data, set size, ordinal or nominal data, data longevity, value sequences, metadata and identifiers. Each of these analysis steps should help narrow down the set of PETs applicable to the given type of data. In our case, this requires quite some effort. We identified 29 data types in our use case (see Table 1), leading to $29 \cdot 6 = 174$ analysis steps. We present here only a couple of those steps as examples.

One criterion is whether the data is continuous or categorical, which poses a challenge since most of our data types (e.g., name, address, vehicle ID, route) are neither continuous nor categorical. Some data (e.g., fare) is continuous. The analysis tells us that some PETs, for example PRAM (post-randomization method), cannot be applied to these data types. Similarly, some of our data (e.g., payment method) is categorical, and the analysis tells us that some PETs, for example noise masking, cannot be applied to these data types. Another criterion is the number of values that the given data type can assume. For most of our data types, this depends on implementation details (e.g., the string length maximally allowed for name or address). This seems to contradict the statement of Kunz et al. that their methodology can be applied in the early phases of the system design process, because such choices may not have been made yet at this stage. Also, Kunz et al. do not specify what to do with this information. They only state that a smaller set of possible values decreases the applicability of PETs. It is not clear how this could help narrow down the set of applicable PETs.

C. Service-driven elicitation. This step entails analyzing the service's requirements on data utility, with the aim of determining which PETs would not undermine the usefulness of the given service. For this purpose, the methodology uses three criteria: value precision, data freshness, and attribute dependency.

As to the first criterion, the "precision required by the service" is unclear for certain data types (e.g., camera feed). For other data types, the precision requirement may vary over time: e.g., the pick-up location must be known exactly when the vehicle picks up the rider, but the precision may be lowered when this data is stored for later processing. Unfortunately, the methodology does not support such varying precision requirements. The second criterion is how fresh the data needs to be. This is again problematic: the same data can be associated with different freshness requirements for different purposes. For example, if the robotaxi encounters a difficult traffic situation and requires remote control from a

human operator, that operator needs the camera feed in real time. On the other hand, for settling compensation claims, there may be a need to access archived camera feeds from weeks before. Again, the methodology does not support this type of varying requirements. The last criterion is the dependency between attributes. Indeed, some of the data types in our use case are not independent. For example, there is a connection between the route and the fare, since a longer route typically leads to a higher fare. Kunz et al. draw our attention to the fact that in such cases, determining different PETs for the dependent attributes may cause problems. It is not clear how this information could help our PET selection process, since the different data types may force us to use different PETs for those attributes. Also, even if the same PET is used for two interdependent attributes, the dependency may still cause problems if not properly taken into account, and the methodology does not clarify how to avoid such problems.

D. PET selection. Assuming that the previous two steps delivered a set of potentially applicable and useful PETs (which is not the case in our use case due to the difficulties reported above), this step aims at choosing the best ones from those sets. Unfortunately, Kunz et al. state that this is highly use-case-specific, so that they do not provide a systematic approach for this step.

Further limitations. As we saw above, steps B and C are only partially applicable to our use case, and step D does not give clear guidance. In addition, the approach suffers from further limitations. First, the approach is limited to data-obfuscation PETs. In our case, several data types (e.g., user name or payment information) must be available to the service provider without modifications for legitimate purposes, so that they cannot be obfuscated. There are data protection requirements associated with these data types, but addressing these requirements requires PETs not supported by the methodology. Second, the approach assumes a list of available PETs. However, finding the right level of abstraction for PETs is challenging. E.g., Kunz et al. consider aggregation to be one PET, but mention that various aggregation techniques exist. Those techniques could be just as well considered individual PETs. If we find out using the methodology that we should use aggregation, we are still faced with the question of which aggregation technique to use. Third, Kunz et al. state that their approach can be used in tandem with LINDDUN. However, the approach excludes two important threats covered by LINDDUN: unawareness and non-compliance. Compliance with data protection regulations is the primary privacy objective for most service providers, making non-compliance the most important threat from their point of view.

5.2 Approach of Kunz and Binder (2022)

Kunz and Binder [21] propose a categorization of PETs to aid PET selection. For each considered PET, they determine the relevant privacy goals, metrics for measuring the PET's privacy effect, the relevant "functional scenario" (one of: release, messaging, authentication, authorization, retrieval, computation), the PET's maturity on a scale from 1 to 3, and the PET's impact on performance, architecture, and utility (the last three are binary attributes: there is either impact or not). The paper provides this categorization for 29 PETs. On this

basis, the following methodology can be deduced. Starting from a privacy threat assessment, first the privacy goal and functional scenario is determined for each threat. Then, the categorization helps identify the subset of PETs applicable to the combination of privacy goal and functional scenario. Finally, the maturity and impact attributes of the short-listed PETs help choose the most appropriate PET. In the following, we go through these steps, applying them to our use case.

A. Identifying privacy goal and functional scenario. A privacy threat assessment of our use case has already been performed by Al-Momani et al. [2] using LINDDUN. The privacy goals used by Kunz and Binder are directly linked to the LINDDUN threat types, which makes it trivial to determine the privacy goal related to each threat. E.g., for a linkability threat, the related privacy goal is unlinkability. Determining the "functional scenario" that provides the context for a threat, however, is not always obvious. Some threats arise in the context of activities that could belong to more than one category: e.g., the threats arising from data sharing between the *SP* and the *OEM* could be seen to belong to both the "release" and the "messaging" category. The functional scenario of some other threats—e.g., the threat of storing personal data beyond its necessary retention period—does not seem to belong to any of the proposed categories.

B. Identifying relevant subset of PETs. If the privacy goal and the functional scenario could be determined for a threat, then the matrix of Kunz and Binder can be used to mechanically determine the subset of relevant PETs. Even this seemingly straightforward step poses difficulties. The matrix offers no PETs for unawareness and non-compliance threats, although, as we mentioned earlier, these threats can be very important. Also, there are many combinations of privacy goal and functional scenario, for which the matrix offers no PETs.

C. Selecting the most appropriate PET. If we managed to identify a set of applicable PETs for a given threat through the two previous steps, then the final step is to select the most appropriate one. Unfortunately, the paper offers no clear guidance on how to do that. It is suggested that the maturity and the impact on performance, architecture, and utility should be helpful in making this decision. But it is not clear how. E.g., suppression and recoding are given as two PETs that can both address linkability threats in a "release" functional scenario, and they have the same maturity and the same impact on performance, architecture, and utility, so it remains unclear which one to choose. Another example: swapping and noise masking can be used for the same type of threat and functional scenario; swapping has a lower maturity than noise masking, but noise masking impacts utility, making it unclear which one to choose.

Further limitations. Beyond the questions that the individual steps raise, the approach also suffers from more general issues. Some are similar to the problems identified in Sect. 5.1. E.g., unawareness and non-compliance are missing in both approaches. Also, we mentioned in Sect. 5.1 that it is difficult to come up with a good list of PETs because it is not clear if different variants of a PET should be regarded as different PETs. For the method of Kunz and Binder, this problem is even more severe because different variants of a PET may have different maturity and different impact on performance, architecture, and utility. E.g., Kunz and

Binder mention synthetic data as a PET. However, there are many ways to generate synthetic data, and their impact on, e.g., utility can be very different.

The impact attributes of Kunz and Binder are problematic anyway. It is not possible to capture the impact of a PET on performance, architecture, and utility in general, because this depends on many further details. E.g., the matrix of Kunz and Binder shows that the PET MPC (multi-party computation) impacts performance. However, there are many MPC techniques, and their performance impact is very different. Even for one particular MPC technique, e.g., additive secret-sharing, its performance impact depends heavily on the types of operations that it is applied to: linear operations (addition or multiplication by a constant) can be very quickly performed on additively secret-shared numbers, whereas non-linear operations are much more costly [27]. Thus, the performance impact depends not only on the PET, but also on the context in which it is applied. A further problem is that the analysis must be performed for every single threat. In a real system, the number of threats can be high, making this impractical. Also, the risk posed by several threats may simply be accepted or may be addressed by non-technical means, so that PET selection for these threats is not necessary. E.g., in our use case, there are obvious identifiability threats stemming from the collected identifiers, but this is accepted because of other requirements. Finally, threats may be connected to each other. The methodology proposes a PET for each threat independently, potentially leading to a sub-optimal solution.

5.3 Approach of Al-Momani et al. (2022)

Al-Momani et al. [3] propose a methodology using decision trees to systematically guide users from privacy threats identified with LINDDUN to suitable privacy solutions. For this, specific key nodes are identified in the LINDDUN threat trees. These nodes contain information regarding the cause of the threat, the threat class, and the system element where the threat applies. For each key node, the mitigation goal is defined, and nodes sharing the same goal are grouped together. In total, ten mitigation goals are defined. For each mitigation goal, potential countermeasures are defined and then ordered according to the data-oriented privacy design strategies [13], i.e., Minimize, Separate, Abstract, and Hide. This process yielded four solution trees for the mitigation goals "protect-attributes", "protect-communication-metadata", "protect-id", and "secure-processing". In the following, we apply this approach to our use case.

A. Identify "key nodes" for the solution trees. To select the applicable PETs, the original approach had to be modified because it had been designed for an earlier version of LINDDUN, rendering the utilization of the key nodes unfeasible. Our adaption process was initiated by mapping the identified threats from the LINDDUN analysis to the solution trees. To maintain a fundamental element of the method—the usage of the rationales underlying a threat identified through the threat trees—we used the assumptions from the use case [2], which encompass analogous information and facilitated the mapping process.

B. Identify possible PETs using the solution trees. The aforementioned new mapping allowed us to use the solution trees, which consequently resulted in

some PETs for the different phases. The first step is to address the applicability of a PET. Then, it is necessary to determine whether the PET alone is adequate to remedy the threat of the key node or if it must be combined with other applicable PETs. In summary, we observed two main outcomes of the method per threat: i) Mitigation is not applicable since the (precise) data is required for the service, e. g. for user identification; and ii) Mitigation is possible using: Remove, Replace, Separate, or use Noisy & less granular attributes, depending on the data.

The proposed solution trees are a promising concept, particularly in terms of prioritizing privacy strategies and assessing the necessity of data. This approach involves determining whether the data is indispensable and, if so, explores options for its replacement, separation, or generalization. Only after this thorough evaluation should the utilization of advanced PETs be considered. However, this method also has major shortcomings. The "*secure-processing*" tree might be complete regarding PETs, since it helps choose one of the three currently available PETs for secure processing: homomorphic encryption, trusted execution environments, and multiparty computation. However, the "*protect-id*" tree considers only attribute-based credentials as a PET which limits usability. The "*protect-attributes*" tree only considers encryption in general and no specific PET. Although the key 'entry' nodes include "Untrusted communication", "Observe message and/or channel", and "Dataflow not fully protected", even TLS is missing as a PET. In addition, technologies that protect attributes are missing, such as attribute-based credentials or zero-knowledge proofs. The "*protect-communication-metadata*" deals with "Non-anonymous Communication" and lists only Onion routing and Hiding timestamps and the message size by random padding as possible PETs.

Further Limitations. The approach suggests primarily to use Hoepman's privacy strategies [13], but lacks more concrete details on PET selection. Missing PETs limit the selection of (advanced) technical PETs.

5.4 A Pragmatic Approach Based on Hoepman (2014)

We now sketch a pragmatic approach based on Hoepman's privacy design strategies [13] and the authors' collective expertise. Al-Momani et al. [2] previously identified the assumptions underlying the privacy threats they found. To address these threats, we revisit their assumptions. We identify the purpose of data processing and explore the potential application of PETs to enhance privacy. Where feasible, appropriate PETs are incorporated.

A. Preparation by applying privacy strategies. Before analyzing the assumptions and phases relevant to PET selection, we adopted the following general strategies (where applicable): i) *Minimize*: We revisited the original assumptions, asking whether the data in question was truly necessary (cf. Sect. 4). For age verification, the application of Attribute-Based Credentials (ABCs) could be considered. ii) *Hide*: Encrypt all collected data at rest (e. g., disk/database encryption) and in transit (e. g., TLS); ii) *Enforce*: Implement strict access control (e. g., role-based) to safeguard data and ensure auditability; iv) *Inform*: Provide users with clear and accessible information about data processing and its purposes, such as through a privacy policy, data collection notices, and regular updates; v)

Control: Enable users to manage their preferences, and access, delete, or update their personal information — via a user dashboard, data deletion protocols, opt-in mechanisms, and consent withdrawal.

B. PET selection process. To identify additional potential PETs, we examined the data items used in each phase. Table 1 provides an overview of how data is used across phases. For example, one result of this activity was the identification of homomorphic encryption as a potential PET for encrypting location, time, and route data of vehicles, thereby enabling vehicle allocation while preserving confidentiality and still allowing matching with the (also encrypted) user location.

C. Threat assessment. We conducted an additional LINDDUN analysis using the revised assumptions. The revised assumptions have the potential to mitigate or eliminate most of the previously identified threats. However, we were unable to eliminate threats regarding linkability and identifiability (LINDDUN threats L.1.1, I.1.1, and I.2.2.1), as these stem from the use of a unique identifier. Nevertheless, for the purposes of our use case, it does not constitute a privacy problem if the *SP* can identify a *User*. It is important to note that even if advanced PETs (e.g., attribute-based credentials, zero knowledge proofs, anonymous payment) are implemented to allow anonymous use of the service, the *SP* may still be able to identify a user through data correlation (e.g., pick-up/drop-off locations, routes, and times), behavioral patterns, or service customization. Furthermore, in certain jurisdictions, the *SP* may be obligated to collect specific information for legal compliance, making full anonymity impossible.

Further Limitations. The main limitation of this approach is that it is not a systematic methodology. We first identified suitable privacy strategies following Hoepman [13], and then mapped them to relevant PETs. However, Hoepman's strategies are defined at a higher level than PET Selection. As a result, we analyzed assumptions and determined the deployability of specific PETs to address certain threats based on our own experience, without a formal method. This introduces two limitations: i) The approach requires experienced experts to produce useful results, and ii) Different teams may reach different conclusions, reducing consistency and repeatability.

6 Analysis of PET Selection Approaches

In this section, we analyze the findings from the three PET selection attempts of Sections 5.1-5.3, highlighting their respective strengths and weaknesses. Table 2 provides a comparative summary of our analysis. We also extract insights to guide future research on PET selection methodologies.

6.1 Strengths

Each of the methodologies considered (Sect. 5.1-5.3) has its own strengths, which are largely complementary.

Table 2: Comparison of PET Selection Approaches

Criterion	Kunz et al. (2020)	Kunz & Binder (2022)	Al-Momani et al. (2022)
Core Method	Data- and service-driven filtering of PETs	PET matrix by goal, scenario, maturity, impact	Decision trees linking LINDDUN threats to strategies
Design Stage Fit	Assumes mature design, known data	Requires detailed threats	Needs mapped assumptions and threats
Final PET Selection Support	No decision logic for choosing among PETs	Maturity/impact noted but no guidance	No prioritization among PETs
Scalability / Use Case Fit	Too granular for large systems	Partial threat coverage	Partial PET coverage; requires expert tuning
Handles Context	Recognizes variation but lacks structured support	Treats PET effects as static across contexts	Accounts for necessity of data
Threat Interdependency	Treats threats independently	Treats threats independently	Considers shared assumptions, but not systematically
PET Coverage	Narrow focus on obfuscation PETs	Moderate PET list with missing types	Incomplete list (e.g., omits TLS, ZKPs, ABCs)
Strengths	Combines data/service analysis; domain-specific taxonomy	Maturity and impact dimensions included	Leverages threat rationale; supports strategy prioritization
Limitations	High effort; limited guidance for final PET selection	Ambiguous threat-to-PET mapping; lacks detail on PET variants	Limited PET set; lacks automation or consistency

The approach of Kunz et al. [20] promotes a combination of data-driven and service-driven elicitation. This is a sensible idea, as both the characteristics of the data and the requirements of the service influence the set of applicable PETs. The paper also introduces the concept of a domain-specific data taxonomy, with a set of applicable PETs mapped to each identified data type. This is an interesting idea that could help make PET selection more efficient.

The approach of Kunz and Binder [21] considers PET maturity as well as the impact of PETs on performance, architecture, and utility. Each of these aspects may be important in practice.

The approach of Al-Momani et al. [3] leverages detailed threat assessment information when selecting PETs. Our experience confirmed the value of this idea: the threat assessment improved our understanding of the origins and potential consequences of privacy threats, which proved helpful for PET selection.

6.2 Weaknesses

As described in Sect. 5, applying each of these academic approaches to our use case was problematic. Beyond the specific weaknesses of individual approaches, which may reflect their relative immaturity, we encountered several recurring limitations that may indicate more fundamental limitations. First, each approach seems to assume a completed system design. However, by that point, introducing PETs may be too late, as they could potentially impact core design choices. None

of the approaches supports an agile process in which the general system design and privacy considerations evolve in parallel, influencing each other iteratively.

Second, each approach assumes a fixed list of PETs and clear criteria for applicability. In practice, PET lists are often arbitrary, and the applicability of a given PET typically depends on context. Determining the impact of a PET (e.g., on performance, architecture, functionality, or future extensibility) requires careful analysis and substantial design effort [23]. The reviewed approaches tend to overlook this and rely on over-simplified generalizations.

Third, while existing approaches may identify potentially applicable PETs, they offer little guidance for making a final selection. This gap is especially critical in scenarios with specific accuracy and performance requirements. For example, when adding noise, it should sufficiently obscure privacy-relevant information without degrading the utility of the data. The performance impact of a PET also depends on the context: real-time applications impose stricter constraints than offline or batch-processing tasks. Moreover, the outcome depends not only on the PET itself but also on its configuration (e.g., the ϵ value in differential privacy).

Fourth, each approach treats threats in isolation, selecting at least one PET per threat. In reality, both threats and PETs may be interdependent. For example, a single PET might mitigate multiple threats, or the use of one PET could interfere with the effectiveness of another. Focusing solely on local decisions can lead to overall suboptimal or even infeasible outcomes.

Finally, each approach omits considerations that fall outside their defined scope, such as "soft privacy" goals or security requirements. While this is understandable in a research setting, practical methodologies must be more comprehensive to be useful in real-world deployments.

6.3 Recommendations for Future Methodology

Insights from the pragmatic approach could help inform the development of improved methodologies. We offer the following recommendations.

Investigate Assumptions. When identifying mitigation techniques, we found it important to trace threats back to their underlying causes. The origin of a threat often constrains the available mitigation options. For example, if Identifiability threats arise due to legal requirements to identify users, then PETs that provide anonymity may not be applicable. To support this process, we found it useful to document data protection-related assumptions about the system and to link each identified threat to the assumptions that give rise to it. This also helped identify cases where multiple threats stemmed from a shared assumption, meaning that a single PET targeting that assumption could address several threats. Revisiting assumptions and clarifying the purpose of data processing proved to be a valuable step in preparing for PET selection.

Specific Step-wise Dataflows. Structuring the use case into discrete steps helped streamline PET selection. It allowed us to visualize when and where data is created, to identify dependencies, and to avoid unintended side effects when applying PETs. A PET applied to mitigate a threat in one step may influence other steps where the same data is used.

PETs' Appropriateness. Addressing the limitations of current approaches will require improved support for selecting PETs in specific scenarios. In particular, new methodologies should help map scenario-specific requirements to the expected changes in system properties (e.g., performance, accuracy) resulting from the implementation and configuration of PETs. This would inevitably bring deployment and integration changes to the system that should be investigated by new methodologies.

Adaption to Design Phase. Different phases of the system design process require distinct tools and approaches. Designing a system from scratch allows building privacy into the architecture from the ground up. In contrast, improving an existing system demands a detailed understanding of current data flows to assess whether introducing a PET is feasible. For example, adding noise to encrypted data is not straightforward and may compromise functionality. Introducing a PET might also disrupt operations if essential data becomes inaccessible. If the system incorporates machine learning, additional considerations arise, such as the distinction between the initial training phase and the deployment of the model, which may affect how and when PETs can be applied.

Addressing Compliance. None of the approaches considered compliance. A future approach for PET selection could aim to bridge the gap between building privacy-friendly systems and ensuring regulatory compliance. Aligning privacy engineering with compliance requirements would significantly improve practical adoption. This is especially relevant in corporate environments, where privacy processes are often structured around meeting legal and regulatory standards.

7 Conclusions and Future Work

The PET selection methods found in the literature exhibit significant shortcomings. While they offer some guidance, they often rely on oversimplified assumptions (e.g., regarding the applicably of a PET in a given situation), and fall short of providing a complete methodology. In some cases, these approaches yield a list of potentially applicable PETs, but the challenge of selecting the most appropriate one remains. This requires evaluating the maturity of each PET, its compatibility with performance and architectural constraints, the availability of ready-to-use implementations etc.

The pragmatic approach presented in this paper cannot be considered a methodology in its current form, as it heavily relies on the expertise of the team. The challenge of selecting appropriate PETs remains open, and current approaches can only partially support this task.

Our work highlights the importance of using realistic use cases for evaluating PET selection methodologies. Post-ride actions, such as service enhancements or monetization, can directly influence PET selection. For example, issuing invoices must comply with legal requirements regarding the included data.

While our analysis highlights the challenges of selecting PETs in real-world scenarios, it does not offer a complete solution. Even after PETs are selected, implementing, integrating, and configuring them remains a significant challenge

[12]. There is a need for more iterative, agile, and exploratory approaches that support "what-if" analysis, allowing design teams to evaluate the impact of selected PETs without immediate commitment. Privacy should be integrated into overall system design, not treated as a separate, downstream process. The use of Artificial Intelligence techniques to support PET selection also represents a potential direction for future work.

Acknowledgments. This work was inspired by privacy engineering discussions at Dagstuhl Seminar 23242, "Privacy Protection of Automated and Self-Driving Vehicles". The work was supported in part by the U.S. National Science Foundation (NSF) under grant number 2245323, and by the German Federal Ministry of Education and Research (BMBF) under grant number 16KIS1382.

Bibliography

[1] Adams, C.: Introduction to Privacy Enhancing Technologies: A Classification-Based Approach to Understanding PETs. Springer (2021)
[2] Al-Momani, A., Balenson, D., Mann, Z.Á., Pape, S., Petit, J., Bösch, C.: Navigating privacy patterns in the era of robotaxis. In: IEEE European Symposium on Security and Privacy Workshops, pp. 32–39, IEEE (2024)
[3] Al-Momani, A., Bösch, C., Wuyts, K., Sion, L., Joosen, W., Kargl, F.: Mitigation lost in translation: leveraging threat information to improve privacy solution selection. In: ACM SAC (2022)
[4] Alhirabi, N., Beaumont, S., Rana, O., Perera, C.: Designing privacy-aware iot for unregulated domains. ACM Transactions on Internet of Things (2023)
[5] Baldassarre, M.T., Barletta, V.S., Caivano, D., Scalera, M.: Integrating security and privacy in software development. Software Quality Journal **28**(3), 987–1018 (2020)
[6] Bellotti, V., Sellen, A.: Design for privacy in ubiquitous computing environments. In: ECSCW, pp. 77–92, Springer (1993)
[7] Chah, B., Lombard, A., Bkakria, A., Yaich, R., Abbas-Turki, A., Galland, S.: Privacy threat analysis for connected and autonomous vehicles. Procedia Computer Science **210**, 36–44 (2022)
[8] Deng, M., Wuyts, K., Scandariato, R., Preneel, B., Joosen, W.: A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. Requirements Engineering **16**(1), 3–32 (2011)
[9] Dev, J., Rashidi, B., Garg, V.: Models of applied privacy (map): A persona based approach to threat modeling. In: ACM CHI, pp. 1–15 (2023)
[10] Drozd, O.: Privacy pattern catalogue: A tool for integrating privacy principles of iso/iec 29100 into the software development process. Privacy and Identity Management pp. 129–140 (2016)
[11] Gürses, S., Troncoso, C., Diaz, C.: Engineering privacy by design. Computers, Privacy & Data Protection **14**(3), 25 (2011)

[12] Herwanto, G.B., Ekaputra, F.J., Quirchmayr, G., Tjoa, A.M.: Towards a holistic privacy requirements engineering process: Insights from a systematic literature review. IEEE Access (2024)
[13] Hoepman, J.: Privacy design strategies - (extended abstract). In: ICT Systems Security and Privacy Protection SEC, IFIP AICT, vol. 428 (2014)
[14] Hong, J.I., Ng, J.D., Lederer, S., Landay, J.A.: Privacy risk models for designing privacy-sensitive ubiquitous computing systems. In: ACM DIS, pp. 91–100 (2004)
[15] Jensen, C., Tullio, J., Potts, C., Mynatt, E.D.: Strap: a structured analysis framework for privacy. Georgia Institute of Technology **1** (2005)
[16] Jordan, S., Fontaine, C., Hendricks-Sturrup, R.: Selecting privacy-enhancing technologies for managing health data use. Frontiers in Public Health **10**, 814163 (2022)
[17] Kalloniatis, C., Kavakli, E., Gritzalis, S.: Addressing privacy requirements in system design: the pris method. Requirements Engineering **13** (2008)
[18] Katcher, S., Ballard, B., Bloom, C., Isaacson, K., McEwen, J., Shapiro, S., Slotter, S., Paes, M., Xu, R.: The mitre panoptic™ privacy threat model tutorial. In: 2nd Workshop on Privacy Threat Modeling (WPTM) (2023)
[19] Kung, A.: Pears: Privacy enhancing architectures. In: Privacy Technologies and Policy - 2nd Annual Privacy Forum (APF), pp. 18–29 (2014)
[20] Kunz, I., Banse, C., Stephanow, P.: Selecting privacy enhancing technologies for iot-based services. In: EAI SecureComm, pp. 455–474 (2020)
[21] Kunz, I., Binder, A.: Application-oriented selection of privacy enhancing technologies. In: Privacy Technologies and Policy - 10th Annual Privacy Forum, APF, LNCS, vol. 13279, pp. 75–87, Springer (2022)
[22] Löbner, S., Tronnier, F., Pape, S., Rannenberg, K.: Comparison of de-identification techniques for privacy preserving data analysis in vehicular data sharing. In: ACM CSCS, pp. 7:1–7:11, ACM (11 2021)
[23] Mann, Z.Á., Petit, J., Thornton, S.M., Buchholz, M., Millar, J.: SPIDER: Interplay assessment method for privacy and other values. In: 2024 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), pp. 1–8, IEEE (2024)
[24] Pape, S., Bkakria, A., Chah, B., Heymann, M., Winkler, S.S.: A framework for supporting PET selection based on GDPR principles. In: ARES (2025)
[25] Pape, S., Rannenberg, K.: Applying privacy patterns to the internet of things' (IoT) architecture. Mobile Networks and Applications (2019)
[26] Pape, S., Syed-Winkler, S., Garcia, A.M., Chah, B., Bkakria, A., Hiller, M., Walcher, T., Lombard, A., Abbas-Turki, A., Yaich, R.: A systematic approach for automotive privacy management. In: ACM CSCS (2023)
[27] de Vries, R., Mann, Z.Á.: Secure neural network inference as a service with resource-constrained clients. In: Proceedings of the IEEE/ACM 16th International Conference on Utility and Cloud Computing (2023)
[28] Wuyts, K., Joosen, W.: Linddun privacy threat modeling: a tutorial. CW Reports (2015)