Addressing Privacy in Software Architecture

Koen Schellens

6608345

Master Thesis

Utrecht University, Princetonplein 5, 3584 CC Utrecht, the Netherlands

Department of Information and Computing Sciences

Supervisors

Dr. Jan Martijn van der Werf

Dr. Fabiano Dalpiaz

June 2021

Abstract

Software architects perform a structural trade-off process to create an architecture that meets the needs of its stakeholders. To embed privacy in this process, it is necessary to formulate privacy as a quality attribute. We propose that the privacy quality attribute consists of three characteristics; the individual, data and unauthorized access. We also propose a set of six architectural tactic groups and a subselection of tactics to work towards an architectural perspective on privacy. The characteristics, tactic groups and tactic selection were formulated by combining a systematic literature review with a grounded theory approach. Afterwards, these artifacts were evaluated by experts in the field of software architecture, privacy, and cybersecurity, who have confirmed the completeness and correctness of our proposed artifacts.

Keywords

Software Architecture, Privacy Engineering, Quality Attributes

Content

Figures

Abbreviations

- CCPA: California Consumer Privacy Act
- CBAM: Cost Benefit Analysis Method
- FIP: Fair Information Practices
- GDPR: General Data Protection Regulation
- LINDDUN: Linkability, Identifiability, Non-repudiation, Detectability, Disclosure of information, Content Unawareness, Policy and consent non-compliance [61]
- NIST: National Institute of Standards and Technology
- PEAR: Privacy-Enhancing Architecture [40]
- PET: Privacy Enhancing Technology
- PG: Protection Goals [32]
- PII: Personally Identifiable Information
- PRIPARE: PReparing Industry to Privacy-by-design by supporting its Application in REsearch [41]
- PS&T: Privacy Strategies and Tactics [11]
- PVD: Privacy Violation Diagram
- QA: Quality Attribute
- SEI: Software Engineering Institute - Carnegie Mellon University

1 Introduction

When the COVID-19 pandemic started spreading across the world in spring 2020, many governments started the construction of contact-tracing apps. In several cases, these hastily developed apps later had to be retracted, because they violated user privacy [16, 28]. These violations were usually not intentional, but merely the result of software design where privacy concerns had not been adequately addressed.

This example illustrates a problem faced by many software development companies; addressing privacy in software development is notoriously complex [30]. Studies indicate there are several reasons for this, ranging from unfamiliarity with the subject to outright opposition to its implementation, when privacy concerns clash with other interests of an organisation [50, 53]. Systematically addressing privacy in software also has not been lawfully required for centuries [22].

This changed in 2016 with the introduction of the General Data Protection Regulation (GDPR). The GDPR made Privacy by Design and by Default mandatory within the European Union and European Economic Area. Since then, organisations risk heavy fines if they do not conform to the privacy requirements set forth in the GDPR [18]. In practice, the influence of the GDPR stretches even beyond EU borders because organisations conducting business with EU or EEA member states have to comply with the GDPR as well [27]. Privacy law is also continuously developed further in other countries such as the United States, its latest addition being the 2018 California Consumer Privacy Act [63].

1.1 Problem statement

The goal of a software architect is to create an architecture that meets the needs of its stakeholders [49]. An essential part of this process is to create a balanced architecture by performing trade-offs between quality attributes, because not all stakeholder needs can always be entirely fulfilled [36]. During the trade-off process, architects use architectural perspectives, which can be described as sets of activities, tactics and guidelines that relate to a specific quality attribute. An architect applies the information in the perspective to modify architectural views to ensure that the system exhibits a higher degree of that quality attribute. By systematically performing this process for all quality attributes relevant to the stakeholder, a balanced architecture can be created.

To support privacy in the architectural trade-off process, it is necessary to create an architectural perspective on privacy, for which privacy first has to be defined as a quality attribute. This study thus aims to embed privacy in the software architecture process, by formulating it as a quality attribute.

The thesis is structured as follows. First, the software architecture principles are introduced in chapter 2 on software architecture. Concepts such as software architecture, views and viewpoints, quality attributes, perspectives, the trade-off process and architectural tactics are described. They provide the necessary framework to develop the privacy quality attribute. Also necessary to develop the privacy quality attribute is chapter 3, where we introduce concepts such as information privacy, Fair Information Practice Principles, Privacy by Design, the General Data Protection Regulation and the Privacy Violation Diagram. When the necessary literary groundwork is done, we propose the privacy quality attribute, consisting of a privacy definition, privacy characteristics and privacy tactic groups in chapter 4. To create the tactic groups, an extensive comparison and selection of privacy tactics from six existing studies has been performed. To evaluate the completeness and correctness of the quality attribute, several expert interviews have been performed, the results of which are described in chapter 5. We conclude this thesis with chapter 6, where we describe conclusions, limitations and do recommendations for future work.

1.2 Research questions

This thesis is built around a main research question and accompanying subquestions, which together aim to provide a solution for the problem statement. The main research question is as follows:

MRQ: How to effectively embed privacy as a quality attribute in software architecture?

The main research question is answered by the following subquestions.

SQ 1. How can privacy be regarded as a quality attribute?

We review what a quality attribute is and determine if privacy can be regarded as one. Studies are examined to determine how a quality attribute can be constructed. We determine whether it is possible to follow this process for privacy.

SQ 2. How can current privacy conceptualizations be harmonized into a quality attribute?

We review current privacy developments and law and relate this to software architecture. We also study six studies containing privacy conceptualizations with the goal of assembling these conceptualizations into a privacy quality attribute.

SQ 3. What is the perceived completeness and correctness of the quality attribute?

We conduct interviews with experts in the areas of software architecture, privacy and cybersecurity to evaluate the proposed privacy quality attribute. We focus on evaluating correctness and completeness of the produced artifacts.

1.3 Methods

We here describe the methods that are used to answer the research questions. As a general structure for this thesis, the design science approach by Wieringa is applied [59]. The design science structure was chosen because it provides us with an evaluated and valid foundation for this type of research.

Using the terminology of Wieringa, we try to improve a certain *problem context* by applying an *artifact*. The problem context is the problem of addressing privacy in software architecture. We try to improve this by proposing the privacy quality attribute artifact. The artifact is then later validated to check whether the artifact will actually improve the problem context.

As Wieringa describes, the cycle of improving the problem context can be performed *ad infinitum*. Due to time constraints, only one iteration of this cycle is performed. The design cycle has the following phases:

1) **Problem investigation**, during which is determined what problem context must be improved and for what reasons.
2) **Treatment design**, when the artifact that could treat the problem context is designed.
3) **Treatment validation**, during which is evaluated whether the problem context is truly improved by application of the artifact.

Figure 1 shows how each research questions is answered by one or more research methods. It also shows the phase of the design cycle related to the research question and shows the deliverable that is produced.

Figure 1: Overview of research subquestions, research methods, design science phases and accompanying deliverables.

Question	Methods	Design Science phases	Deliverable
MRQ. How to effectively embed privacy as a quality attribute in software architecture?	*All below*	1) Problem investigation 2) Treatment design 3) Treatment validation	Evaluated privacy quality attribute

SQ 1. How can privacy be regarded as a quality attribute?	Literature Review	1) Problem investigation	Quality attribute background
SQ 2. How can current privacy conceptualizations be harmonized into a quality attribute?	Systematic Literature Review, Grounded Theory	2) Treatment design	Privacy quality attribute
SQ 3. What is the perceived completeness and correctness of the quality attribute?	Expert Interviews	3) Treatment validation	Evaluated privacy quality attribute

The following methods are used to answer the research questions.

1.3.1 Literature review

To create a comprehensive overview of software architecture and privacy, we examine key literature in these research areas. We judge to what extent privacy fits in the software architecture paradigm. We use this information to answer subquestion 1. The information is summarized in chapters 2 on software architecture and 3 on privacy.

1.3.2 Systematic Literature Review

A systematic literature review (SLR) is performed to answer subquestion 2. An SLR is used to summarise all existing information about a phenomenon in a thorough, unbiased and reproducible manner [37]. The goal of our systematic literature review is to gather relevant privacy conceptualizations, later to be used to create the privacy quality attribute. The following selection criteria were applied:

1. **Recent.** Only studies published after 2012 are included because of the impact of the GDPR, of which a first draft was published early 2012 [17].

2. **Applicable.** Only studies that are aimed towards conceptualizations of privacy used in software engineering methods or privacy engineering are included. Most studies include both.
3. **Topical**. Only studies that present their most recent privacy conceptualizations or privacy engineering methods are included. If conceptualizations or methods extend previous comparable studies, only the most recent study is included.

The full review protocol can be found in the method section of chapter 4, section 4.4.1.

1.3.3 Grounded Theory

In conjunction with the systematic literature review, we apply grounded theory aspects to answer subquestion 2. The systematic literature review presents us with an extensive list of privacy conceptualizations and architectural tactics. To order and categorise these tactics, we apply a grounded theory approach.

Grounded theory methods are used to create a consistent set of categories of a certain type of data [8]. They are used to create new theory (grounded theory) by collecting and iteratively analysing qualitative data. In this case, the qualitative data consists of the architectural tactics that are produced by the SLR. We thus somewhat deviate from the standard grounded theory path, because most of the data collection is performed up front. Then, through coding and analysis activities, the tactics are categorized, leading to categorisation and definition of properties. When starting out with analysis and coding activities, many new categories tend to arise. After a while, these categories get saturated and may be grouped to create a definitive set of categories. The coding and analysis steps taken are described in more detail in section 4.4.1.

We are aware that our approach does not fully follow all aspects of standardized methods by, for example, Glaser or Strauss. During the data collection stage for example, it is customary to gather this data from interviews or from observations [8]. We however consult literature as data source. Another difference is the coding process, which we do only partly. The open coding phase is not performed as such, but axial coding and selective coding activities have been performed to create and select categories. All together, we reckon that our approach uses core concepts of grounded theory and we therefore consider it as a form of grounded theory. However, it may not fully match existing grounded theory methods.

1.3.4 Expert Interviews

To answer subquestion 3 and evaluate the artifact, expert interviews are conducted. We are interested in evaluating our proposed privacy quality attribute, privacy characteristics and privacy tactic groups, by checking for completeness and correctness of these artifacts. The interviews have a semi-structured form to allow for discussion and open dialogue.

We note that completeness and correctness does only partially evaluate our artefacts. To make the evaluation more complete, an evaluation could be done that includes whether the artefact improves the way software architects address privacy in their architectures.

The structure of the interviews is described in more detail in section 5.1. The results of the interviews are described per participant in section 5.2 and the discussion of the interviews can be found in section 5.3.

2 Software Architecture

This chapter describes common definitions of software architecture, specifies terminology used within the field, and explains how a quality attribute like privacy can possibly be addressed by applying software architecture principles.

2.1 Defining software architecture

Software architecture refers to the fundamental structures of a software system and the discipline of creating such structures and systems [10]. It is a subdiscipline within the research field of software engineering [39]. The goal of a software architect is to create an architecture that meets the needs of its stakeholders [49].

Two definitions are generally used to describe the software architecture discipline; the set of structures definition by Bass et al., and the set of decisions definition by Jansen and Bosch.

2.1.1 Set of structures

The set of structures definition was introduced by Bass et al. [3]. According to Bass et al., "The software architecture of a system is the set of structures needed to reason about the system, which comprise software elements, relations among them, and properties of both.". Structures are comprised of software elements, relations among them, and properties of both elements and relations [10]. These structures are usually visualized by creating diagrams that emphasize certain aspects of these structures. Examples are the UML Class Diagram and the Functional Architecture Model [4].

2.1.2 Set of decisions

Jansen and Bosch have proposed an additional definition of software architecture, because they perceive several problems when software architecture is approached as just a set of structures. They state that the high cost for change, complex nature and susceptibleness to architectural erosion of a set of structures are partly due to knowledge vaporization [35]. Knowledge vaporization occurs when architectural decisions are only implicitly embedded in the architecture, which is the case when the architecture of a system is solely expressed through diagrams.

To solve this issue, they present a new approach where software architecture is perceived as a composition of a set of explicit design decisions. By making design decisions explicit, theory is, knowledge vaporization is reduced and the long-term value of the software architecture is better preserved. This can be done by textually describing the architectural decisions that lead to the eventual architecture description.

2.2 Views and viewpoints

To visualize a software architecture, its structure is often represented by one or more models or diagrams [49]. A common pitfall is to try to include all aspects of the architecture in a single diagram, which tends to lead to a heavily convoluted diagram that does not meet its goal: to represent a complex system in a manageable and comprehensible way so that it can be reasoned about [10]. To address this issue, a software architecture can be represented by several views that together describe the whole architecture.

2.2.1 Views

The principles of views were first pioneered by David Parnas in 1972 [45]. It has since been improved and expanded upon in several studies, for example in the 4+1 view model by Philippe Kruchten [38]. A contemporary definition based on IEEE Standard 1471 is given by Rozanski and Woods: “A view is a representation of one or more structural aspects of an architecture that illustrates how the architecture addresses one or more concerns held by one or more of its stakeholders.” [49].

Views are aimed at highlighting specific elements of an architecture, to demonstrate to stakeholders that their concerns have been met. In these views, both the functional requirements and the quality attributes of a system can be represented. This can be done by applying architectural perspectives, which is further addressed in section 2.3.

Examples of views are Kruchtens physical view [38], used to describe the mapping of software onto the hardware, or the concurrency view by Rozanski and Woods, which describes the concurrency structure of the system.

Within the same view “category”, different viewpoints exist to highlight different elements of a view. For example, one viewpoint may be used to show in what order network requests are performed, while another viewpoint may be used to show how a systems’ internal components work together. Together, these viewpoints form a concurrency view.

2.2.2 Viewpoints

The goal of a viewpoint is to create a framework for capturing reusable architectural knowledge that can be reused to create an architecture design [49]. Rozanski and Woods have again taken the IEEE standard and define a viewpoint as follows: “A viewpoint is a collection of patterns, templates and conventions for constructing one type of view. It defines the stakeholders whose concerns are reflected in the viewpoint and the guidelines, principles and template models for constructing its views.”.

Viewpoints have the advantage of bringing structure and consistency to the software architecture process. Most software projects use different notations for their architecture designs, although the introduction of commonly accepted languages like UML have led to an improvement on the quality of architecture designs in the past years [49].

2.3 Quality attributes

Views and viewpoints are not just used to describe the structures of a system, but can also be used to express the quality attributes that a system has to exhibit. Quality attributes are also known as cross-cutting-concerns or non-functional requirements. They are used to describe certain properties that a system needs to exhibit, like security or performance [49].

The ISO 25010 standard determines the following regarding the quality of a system [62]:

> *"The quality of a system is the degree to which the system satisfies the stated and implied needs of its various stakeholders, and thus provides value."*

According to the same ISO 25010 standard, a quality attribute like security can be specified by formulating it as a quality model. A quality model shows the characteristics, subcharacteristics and quality properties that make up the quality. Thus follows, if security is to be regarded as a quality attribute, its characteristics, subcharacteristics and quality properties need to be defined. A partial example for the security quality attribute can be found in the figure below.

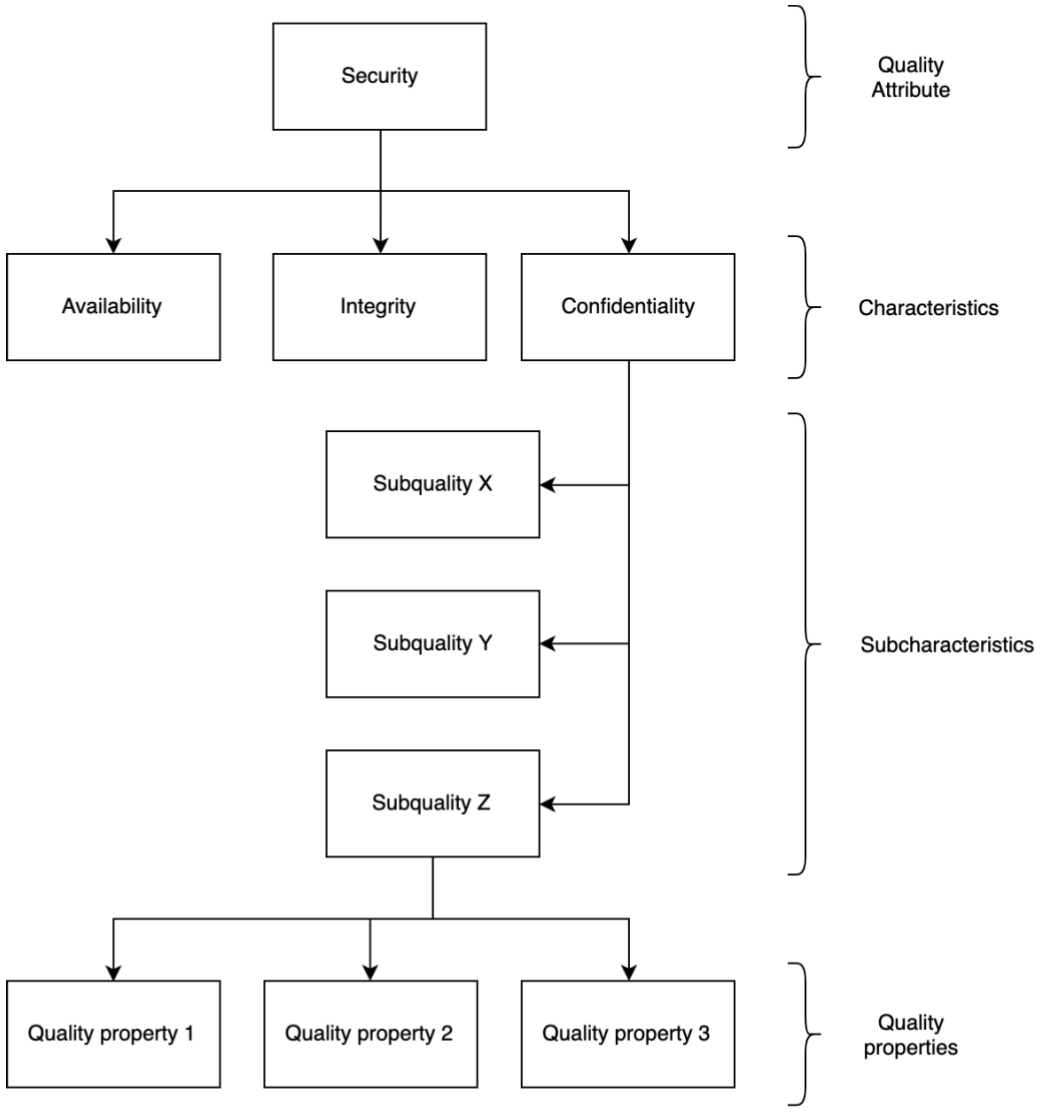


Figure 2: Partial example of the security quality attribute, showing the confidentiality, integrity and availability characteristics.

2.4 Perspectives and trade-off process

A common way of addressing quality attributes in a software architecture is through architectural perspectives. Architectural perspectives can be described as a collection of architectural activities, tactics and guidelines which are used to ensure that the system exhibits the preferred quality properties [49]. This is done by applying the perspective to a relevant view, illustrated in
Figure 3. By applying a perspective to a view, the architect is guided through the process of analysing whether a certain view will lead to the desired quality property.

For example, when a security perspective is applied on a concurrency view, the architect may realise that appropriate precautions against man-in-the-middle attacks are not included in the view. The architect then acts upon this fact by introducing, for example, a certificate-based authorization pattern which mitigates the danger of a man-in-the-middle attack. By including this information in the view, the desired security quality property has been improved. The systematic application this process for all relevant quality attributes is called the architectural trade-of process.

A description of a perspective by Rozanski and Woods [48] typically contains the following sections:

1. **Applicability to views.** Determines on which views the perspective can be applied.
2. **Concerns.** Describes which concerns are addressed by the perspective.
3. **Activities**. Describes which activities are to be performed when applying the perspective to the architecture.
4. **Architectural tactics**. Describes which architectural tactics can be considered to satisfy the required quality properties.
5. **Problems and pitfalls**. Describes common problems and pitfalls to be aware of. Also describes which techniques exist to mitigate the risk.
6. **Checklists**. Optional. Describes things to consider when applying the perspective that are not covered by previous sections.

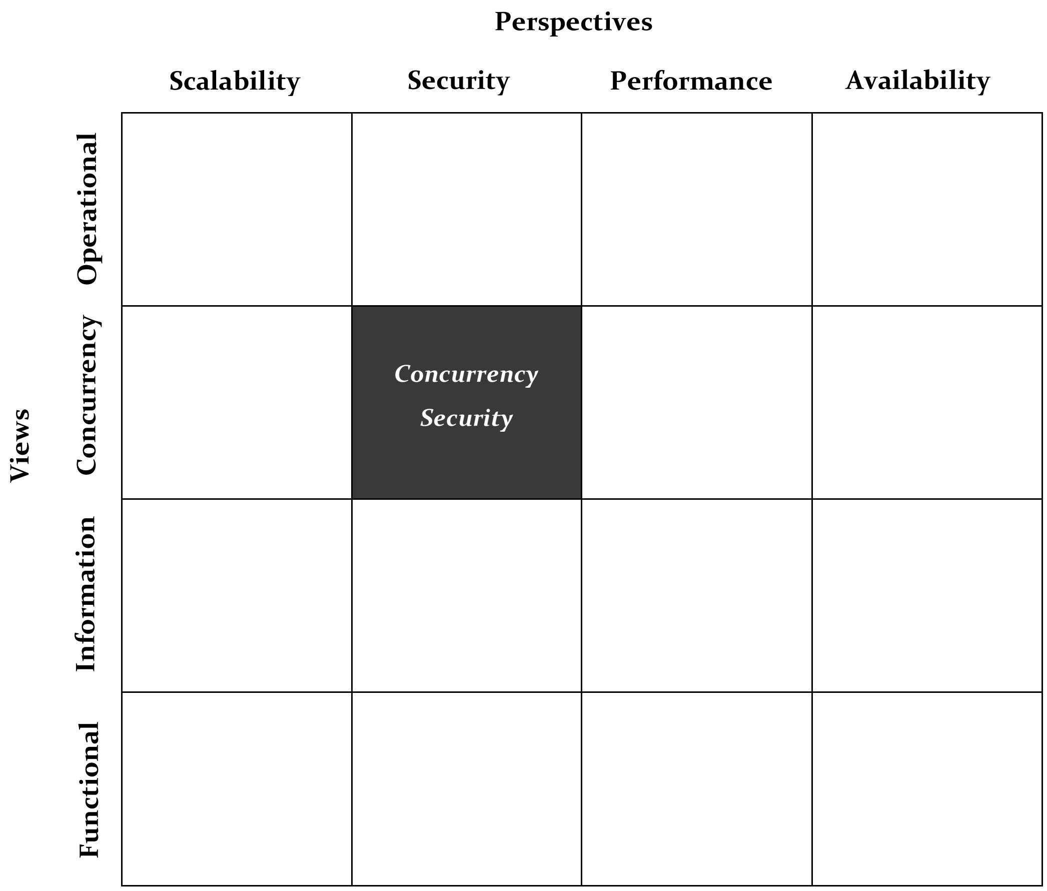


Figure 3: Mapping the security perspective on the concurrency view.

2.5 Architectural strategies, tactics, styles and patterns

In this section, we give a short overview of the differences between architectural strategies, tactics, styles and patterns. Even among studies and books of comparable authoritativeness, we find disagreements with regards to terminology when it comes to defining these concepts. We therefore deem it necessary to specify the definitions that we use in this thesis and provide an explanation for each choice.

Architectural strategies. Colesky et al. propose privacy design strategies as a layer above architectural tactics, which they state is a layer above patterns [11]. However, because strategies do not seem to be a commonly accepted term in software architecture literature, we choose not to not use this definition. Architectural strategies are mentioned by Nord et al. [2]

and Bass et al. [3] in the context of the Cost Benefit Analysis Method (CBAM), but only in a generic way that is not clearly defined.

We do however share the need of Colesky et al. to group architectural tactics with a similar goal. We therefore choose to use the term "architectural tactic groups", or "tactic groups" in short, because architectural tactics are defined more clearly in literature.

Architectural tactics. We use the definition proposed by Rozanski and Woods, which is similar to the definition used by the SEI [1]. "An architectural tactic is an established and proven approach you can use to help achieve a particular quality property." [49]. Architectural tactics can be used to address characteristics of the quality attribute of a system. "They provide the architect with advice on how to address a general issue. It may involve the application of one or more design patterns, but it need not, and it could provide much more general advice.".

Architectural patterns. The structure of an architectural pattern as proposed by Bass et al. uses the principles of a *context*, *problem,* and *solution* [3]**.** Problematically, Rozanski and Woods use the same structure for what they call an **architectural style** [49]. In this thesis, we do not use the concepts of architectural patterns or architectural styles; we focus just on architectural tactics, which are more abstract.

Design patterns. Rozanski and Woods also use the term "pattern", but in the context of a software design pattern, which is "structure of interconnected design elements that solves a general design problem within a particular context". A software design pattern is focussed on the detailed design of the system. It guides the software designer to organise software design units such, as classes and procedures [49]. We again do not use the design pattern concept in this thesis, but only focus on architectural tactics.

In summary, in this thesis we mainly focus on **architectural tactics** as described by Rozanski and Woods. We use **architectural tactic groups**, or **tactic groups** when we need to group architectural tactics with a similar goal.

2.6 Conclusion

A software architecture can be defined as a set of structures and as a set of decisions [3, 35] . The set of structures is visualized by creating architectural models and descriptions, which can be part of a view or viewpoint [60]. The set of decisions is a textual description of the decisions that support the architecture design [35]. To address quality attributes, perspectives can be applied on views to address concerns [3]. If privacy can be regarded as a quality attribute, it could be addressed in software architecture like other quality attributes, leading to an architecture design where privacy is included in a structured and systematic way. Several concepts related to strategies, tactics and patterns exist in software architecture literature to create an architecture. To be as clear and concise as possible, we only use the terms architectural tactics and architectural tactic groups in this study.

3 Privacy

This section aims to provide an up to date overview of information privacy practices in law, which are a necessary basis to later interpret privacy conceptualizations and methods. The section starts with a short introduction on privacy history. We also explain the distinction between privacy and information privacy, followed by background information on the principles of Privacy by Design (PbD) and the EU General Data Protection Regulation (GDPR). Note that this study does not limit itself to the specifications of the GDPR. However, the GDPR is explained in more detail due to its impact on recent privacy studies.

3.1 Historic development

The first attempt at conceptualizing privacy was made in 1890 by U.S. law scholars Brandeis and Warren [57]. In their law review article, "The Right to Privacy", Warren and Brandeis defined privacy as the "right to be let alone". Since then, several amendments have been made to facilitate new insights and developments.

A prominent definition of privacy that is still widely used today, was proposed by law scholar Alan Westin in 1967: "Privacy is the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others." [9]. Westin was the first to define privacy as the ability to control how much about ourselves we reveal to others [24].

The introduction of the Fair Information Practices in 1973 is another development that is still relevant to this day [47]. They were proposed by the U.S. Department of Health, Education & Welfare to protect personal data in record-keeping systems [26]. The Fair Information Practices (FIPs) were originally not legally binding; they were proposed as best practices. However, parts of these practices are still partially or fully reflected in U.S. and EU law. An example of this is the original fourth Fair Information Practice, stating that "there must be a way for a person to correct or amend a record of identifiable information about the person." [55], which shows parallels with Article 16 of the GDPR (Right to Rectification) [56].

The original version of the FIPs from 1973 contains the following five principles [55]:

1. There must be no personal data record-keeping systems whose very existence is secret.
2. There must be a way for a person to find out what information about the person is in a record and how it is used.
3. There must be a way for a person to prevent information about the person that was obtained for one purpose from being used or made available for other purposes without the person's consent.
4. There must be a way for a person to correct or amend a record of identifiable information about the person.
5. Any organization creating, maintaining, using, or disseminating records of identifiable personal data must assure the reliability of the data for their intended use and must take precautions to prevent misuses of the data.

Several versions and adaptations of the FIPs have been created since their introduction [33]. In 1981, the Council of Europe introduced the *Convention for the protection of individuals with regard to automatic processing of personal data*, which strongly reflected the principles. This convention was ratified by both members of the Council of Europe and by several non-European countries [19].

It is necessary to determine to which extent privacy policies are legally binding, because this largely determines how they are being acted upon by businesses and other organisations. As the name suggests, the 1981 *Guidelines on the Protection of Privacy and Transborder Flows of Personal Data* are only guidelines, much like the FIPs from the 1970s.

Therefore, in 1995, the European Union took a more coordinated approach with the introduction of the Data Protection Directive [56]. This directive is legally binding to EU member states, requiring them to achieve the goals or results determined in the directive, but without specifically instructing them how to do so. Member states introduced national laws to achieve the goals set forward in the directive, a process called transposition [20].

This approach changed again with the introduction of the 2016 General Data Protection Regulation. The GDPR is an EU regulation, meaning that no transposition is needed to carry out the regulation; it is immediately enforceable as law in all member states [25]. The GDPR is explained in more detail in section 3.4.

3.2 Information Privacy

Privacy is a multifaceted concept [52]. It therefore is necessary to determine which aspect of privacy is meant whenever we mention privacy.

Privacy experts tend to agree that several categories of privacy exist, however, they agree much less on what these categories should look like [15]. We do not delve too much in these categorisations. However, a common categorisation is that of Burgoon from 1982, where she categorises privacy in information privacy, social privacy, psychological privacy and physical privacy [5]. She defines information privacy as follows: “Informational privacy measures how far people can decide what information is collected about themselves” [5]. A more recent but comparable definition of information privacy is provided by Solove in 2006: “Information privacy is […] not only about controlling immediate access to oneself, but also about reducing the risk that personal information be used in an unwanted way.” [51]. This definition is largely in line with the definition proposed by Westin [9]. We zoom in further on these definitions and its implications in section 3.5 on privacy violations.

This study focusses on software systems and software architecture. Therefore, when privacy is mentioned, this should be read as information privacy except when specified otherwise.

3.3 Privacy by Design

A formalized report on *Privacy by Design* was published by Ann Cavoukian in 1995 in collaboration with the Dutch Data Protection Authority (AP) and Netherlands Organisation for Applied Scientific Research (TNO). Privacy by Design attempts to mitigate privacy concerns through the application of seven Foundational Principles [7]. These principles are:

1. Proactive not reactive - Preventative, not Remedial.
2. Privacy as the default.
3. Privacy Embedded into Design.
4. Full functionality - Positive Sum not Zero Sum.

5. End-to-end security - Lifecyle Protection.
6. Visibility and Transparency.
7. Respect for User Privacy.

A common critique on the original document is its recursive nature; as Gürses and Troncoso point out, "Privacy by design means applying privacy by design – [this] communicates to the reader that something needs to be done about privacy from the beginning of systems development, but it is not clear what exactly this privacy matter is nor how it can be translated into design." [30]. This passage refers specifically to the second principle, "Privacy as the default". As a result, the precise meaning of Privacy by Design has become multi-interpretable: the name suggests that Privacy by Design solely refers to the concept that privacy should be implemented from the start, while it encompasses many more principles. In literature, it is not always clear which explanation is meant.

In its *Preliminary Opinion on Privacy by Design*, the European Data Protection Supervisor notes that data protection by design and by default now have become a legal obligation in nations where the GDPR applies [22]. This is expressed in Article 25 of the GDPR, titled "Data protection by design and by default", which instructs data controllers to implement appropriate technical and organisational measures to safeguard privacy, both at design and operational phase. Several elements of the *Privacy by Design* report have thus found their way into the GDPR, introduced in the next section.

3.4 General Data Protection Regulation

In 2016, the European Union introduced the General Data Protection Regulation (GDPR) [22]. The GDPR made privacy by design and by default mandatory for organisations situated within or dealing with organisations within the EU and EEA [27]. The core concepts of the GDPR are as follows.

Personal data. 'Personal data' means any information relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person [56]. In the USA NIST Privacy Framework, this type of data is called Personally Identifiable Information (PII) [42].

Data Controller or Data Processor. The GDPR determines that an organisation can either be a data controller or processor. The difference is that the data controller determines for which purpose and by which means personal data is processed, while a data processor only processes data on behalf of the controller [13]. "The duties of the processor towards the controller must be specified in a contract or another legal act. A typical activity of processors is offering IT solutions, including cloud storage. The data processor may only sub-contract a part of its task to another processor or appoint a joint processor when it has received prior written authorisation from the data controller." [56]

Records of Processing Activities. Organisations have to keep records of the data processing activities they perform. These records should contain information about the purpose of the information processing, the type of data that is processed and a description of the technical and organisational security measures that are applied [56].

Data Protection Officer. If the core activities of a private organisation consist of regular and systematic monitoring of data subjects, or if special categories of personal data are processed, these organisations are obliged to appoint a Data Protection Officer (DPO). It is the task of the DPO to ensure that the organisation does indeed carry out its responsibilities with regards to the GDPR [56].

Data Protection Impact Assessment. If data processing activities have a high risk of harming the rights and freedoms of the data subjects about which the data is processed, a Data Protection Impact Assessment (DPIA) must be carried out to identify potential risks and create appropriate measures to mitigate these risks [56].

Data Protection by Design and by Default. For a software engineer, an important GDPR requirement is the concept of Data Protection by Design and by Default [30]. It should be noted that Data Protection by Design and by Default is not the same as Privacy by Design by Ann Cavoukian [7], although elements from Privacy by Design have made it into the Data

Protection Directive from 1995 and into the GDPR as well [56]. The essence of Data Protection by Design and by Default, article 25 of the GDPR, determines that "the controller of the data shall implement appropriate technical and organisational measures, both at the design phase of the processing and at its operation, to effectively integrate data protection safeguards to comply with the Regulation and protect the fundamental rights of the individuals whose data are processed." [23]. This obligation results in four dimensions, as formulated by the European Data Protection Supervisor [22]:

1. The processing of personal data by an IT system should always be the outcome of a design project. Article 25 [...] requires that data safeguards are considered during the design phase and operational phase of said project, to ensure safeguarding of data during the whole project lifecycle.
2. A risk management approach should be adopted with regards to selecting and implementing protection measures. The assets to be protected are the individuals of whom the data is processed.
3. The selected protection measures should be appropriate and effective. The measures should be able to demonstrate GDPR compliance, should implement data protection principles and should protect the data of the individual.
4. The selected safeguards should be implemented.

Data Subject Rights. Individuals of whom data is processed have several rights with regards to how their data is processed by others. Data subjects may for example request processed information, request to erase personal data and rectify incomplete or incorrect personal data [56].

Data Breach Notification. In the case of a data breach, the data controller should notify the local Supervisory Authority within 72 hours after it became aware of the breach. The Supervisory Authority differs by member state [56].

Special Categories of Personal Data. The GDPR specifies special categories of personal data, such as political preference, religious or philosophical beliefs and health information. These categories of data are considered high risk and are thus subject to more stringent protection mechanisms [56].

The requirements mentioned above capture several essential requirements but do not provide a full overview of the GDPR. The full text of the GDPR can be viewed online at https://eur-lex.europa.eu/eli/reg/2016/679/oj.

3.5 Privacy violations

As mentioned in section 3.1, Westin has defined privacy as the claim of individuals, groups or institutions to determine for themselves when, how and to what extent information about them is communicated to others [9]. This reasoning is supported by other authors such as Solove. He notes that information privacy is not only about controlling immediate access to oneself, but also about reducing the risk that personal information be used in an unwanted way [54].

We note that the privacy definitions of Westin and Solove share three common aspects: the individual, data about the individual, and preventing unauthorized access to this data. These notions are reflected in the ISO 29100 definition of a privacy breach: a "situation where personally identifiable information is processed in violation of one or more relevant privacy safeguarding requirements". [14]. To visualize these common aspects, we propose the Privacy Violation Diagram.

3.5.1 Privacy Violation Diagram

The Privacy Violation Diagram (Figure 4) visualizes the three main aspects of a privacy violation and demonstrates the presumed interdependency between these three aspects. We hypothesize that a privacy violation requires the intersection of the following: an **individual**, **data** and **unauthorized access**. We reason that all three of these aspects need to be present for an action to be considered a privacy violation.

Individual (1)

We propose the first required aspect for a privacy violation to be a human individual. We reason that without an individual, there is no one whose privacy can be violated. We define an individual as that which exists as a distinct human entity. This is semantically in line with Cambridge Dictionary's definition [6], but rephrased to be more concise.

Data (2)

To define the data aspect, we use the definition provided by the OECD: "Data are characteristics or information, usually numerical, that are collected through observation." [44] Data can be represented both by digital and analogue means. Where the data aspect intersects with the aspect of the individual, attention should be paid to the kind of data that intersects. If the data consists of Personally Identifiable Information (NIST) or Personal Data (GDPR), the data can be used to identify the individual, directly or indirectly by combining multiple quasi-identifiers.

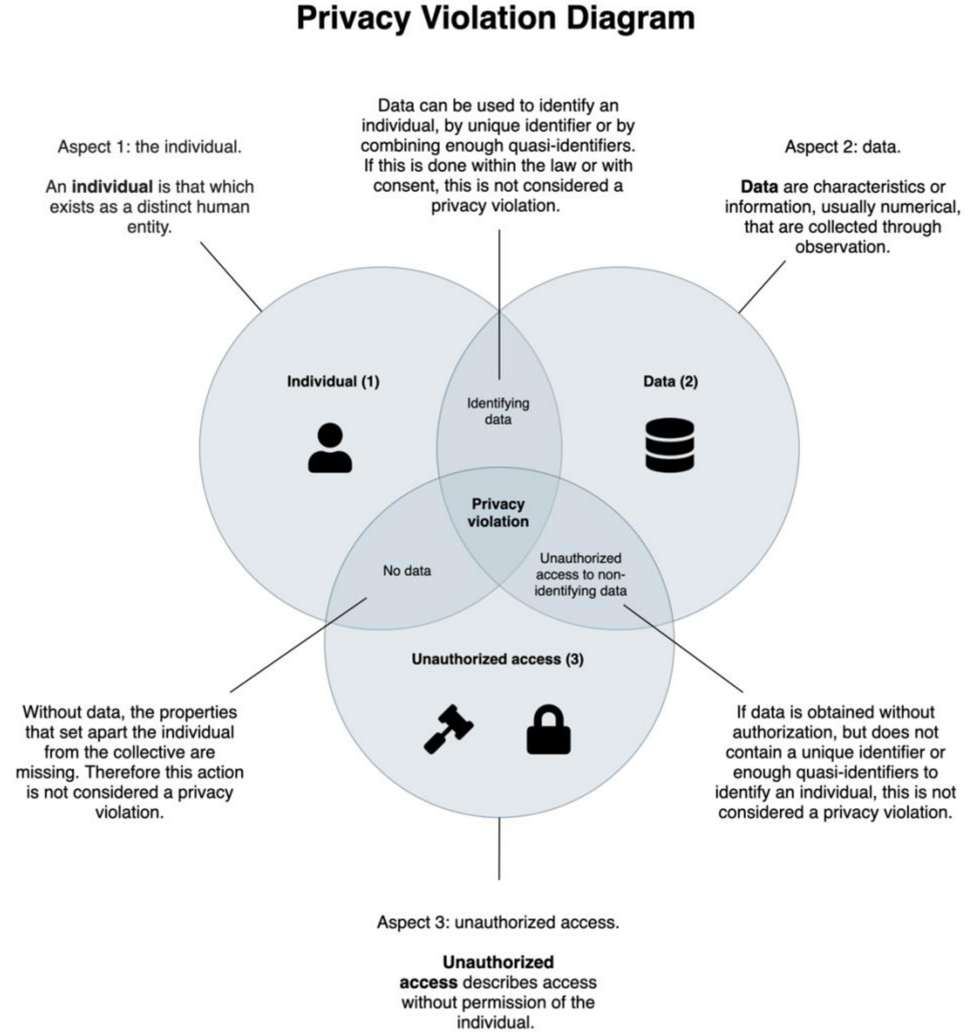

Privacy Violation Diagram
Aspect 1: the individual.
An individual is that which exists as a distinct human entity.
Data can be used to identify an individual, by unique identifier or by combining enough quasi-identifiers. If this is done within the law or with consent, this is not considered a privacy violation.
Aspect 2: data.
Data are characteristics or information, usually numerical, that are collected through observation.
Individual (1)
Identifying data
Data (2)
Privacy violation
No data
Unauthorized access to non-identifying data
Unauthorized access (3)
Without data, the properties that set apart the individual from the collective are missing. Therefore this action is not considered a privacy violation.
If data is obtained without authorization, but does not contain a unique identifier or enough quasi-identifiers to identify an individual, this is not considered a privacy violation.
Aspect 3: unauthorized access.
Unauthorized access describes access without permission of the individual.

Figure 4: Privacy Violation Diagram

Unauthorized access (3)

Unauthorized access describes access without permission of the individual whom the data is about. Authorization can be given by the individual by providing consent to the organisation that processes its data.

We have considered to implement local legislature in the unauthorized access characteristic. As Solove notices, privacy is a culturally fluid concept that varies by country [52]. Thus, a certain degree of variability exists when it comes to what privacy entails. By accepting this variability and assuming it is codified in law, we could argue that it is right and practical for the Privacy Violation Diagram to reflect this variability. We however decided against this implementation, because stating that the characteristics of a privacy violation are dependent on local law can be problematic, because law can be changed by the legislature. This would mean that the characteristics of privacy would be subject to change as well, which can have unforeseen consequences for the interpretation of the diagram. We therefore have decided not to link law or legislation to the PVD.

Intersections

To illustrate how the PVD can be used in practice to determine a privacy violation, we consider the following scenarios. The three aspects of the model are indicated as a number between brackets.

1. In this intersection, the individual (1) intersects with the data aspect (2), because an employee views data about an individual in a CRM system. The access gained to the data is authorized, because the individual has given consent for its data to be viewed by the employee. Therefore, we do not consider this to be a privacy violation.
2. In the second intersection, a system is hacked and a dataset (2) is illegally procured by a hacker group (3). However, because of pseudonymization or encryption of the personal data in the dataset, the data cannot be used to identify any individuals in the set (1). Therefore, we do not consider this a privacy violation, but “merely” unauthorized access.
3. For the third intersection, unauthorized access is obtained (3) which intersects with the abstract concept of the individual (1), but no data (2) is at stake. From the

perspective of privacy, this situation is hard to imagine. The reason for this is that, without data, an individual cannot be distinguished from the collective, because the properties that set the individual apart are missing from the equation. Therefore, we do not consider this to be a privacy violation.

4. Lastly, we present an example where all aspects overlap. A dataset (2) containing personal data (1) has accidently been made public by a data controller (3). The individuals in the dataset have not given their permission for their personal data to be made public. We therefore consider this to be a privacy violation.

The proposed Privacy Violation Diagram may not be an exhaustive overview of privacy violation aspects. Yet, because of its simplicity and seeming exhaustiveness, we use it further on in this thesis to further analyse privacy conceptualizations and to illustrate how their strategies, tactics and patterns address different parts of privacy.

3.6 Conclusion

Privacy is a constantly evolving concept, safeguarded differently in different countries. There is a trend towards more stringent privacy protection laws. Privacy is usually protected through data protection, for example in the GDPR. To safeguard privacy in software systems, it is essential that privacy considerations are taken into account from the start. These principles are usually referred to as Privacy by Design. We propose that a privacy violation consists of three aspects; the individual, data, and unauthorized access. We propose the Privacy Violation Diagram to visualize these aspects.

4 Privacy as a Quality Attribute

In this chapter, we make a step to consider privacy as a quality attribute. We thereby aim to answer subquestion 2: How can current privacy conceptualizations be harmonized into a quality attribute? To answer this question, we need to know the structure of a quality attribute. This is introduced in the first section of this chapter. We then propose a definition of privacy, privacy characteristics, and architectural privacy tactics.

4.1 Structure of a quality attribute

As introduced in section 2.3, a quality attribute contains a definition, characteristics, and tactics [62]. These can be defined as follows:

Definition. The definition of a quality attribute summarizes the quality attribute, provides boundaries and success factors and sets expectations (section 4.2).
Characteristics. A quality attribute includes characteristics that together form the quality attribute. The characteristics can be addressed by architectural tactics and measured by applying quality scenarios (section 4.3).
Tactics. These architectural tactics are best practices to solve common issues of the quality attribute in an architecture definition (section 4.4).

4.2 Definition of privacy

As a definition, we use the definition by Alan Westin: “Privacy is the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others.” [9]. We deem this definition fit for use in software architecture, because it perceives privacy through the lens of information. Software systems are in essence systems that process information, we therefore think Westin’s definition is a fitting definition and a useful way to scope the quality attribute. The definition also is in line with the characteristics of privacy that we propose in the next section.

4.3 Privacy characteristics

To determine the characteristics of privacy, we have to make a clear distinction between quality attribute characteristics and architectural tactics. We then propose our set of privacy of characteristics based on existing literature and legislation.

4.3.1 Characteristics vs. tactics

It is necessary to strictly determine the differences between characteristics and tactics, because in the methods we studied, the distinction was not always clear. Variants of these concepts are respectively called: privacy properties [61], privacy objectives [42], privacy design strategies [11], privacy principles [41], privacy protection goals [32] or privacy tactics [40]. To avoid confusion, we make a clear distinction between the two.

A quality attribute consists of characteristics that together make up the quality attribute [62]. Architectural tactics can then be used to improve these characteristics [48]. By applying tactics, different parts of the quality attribute can be improved, thereby improving the quality attribute in an architecture. Thus, characteristics can be seen as the "what" that the quality attribute consists of, and tactics as the "how" that these characteristics are improved.

As mentioned, in privacy engineering studies, characteristics and tactics tend to get mixed up. In the LINDDUN study, privacy tactics such as unlinkability are presented as privacy properties [61]. We argue however that the LINDDUN privacy properties are ways of improving privacy (tactics), rather than being properties of privacy itself (characteristics). Comparable problems are found in studies introduced in the next section.

We argue that the large majority of the tactics in the six studies in this thesis are different ways of improving the concept of privacy as a whole, rather than them being privacy characteristics. They may be usable to address privacy as a whole concept, but they are not clear in what properties of privacy they precisely address. Therefore, we need a different approach to determine the characteristics of privacy.

4.3.2 Determining characteristics

We propose the three aspects of the Privacy Violation Diagram introduced in section 3.5 as the characteristics of the privacy quality attribute. A limitation of this proposal is that this approach conceptualizes privacy by determining the characteristics of the antithesis of privacy; the violation of it. Our defence to this problem is that the same is generally done for security, which is an established quality attribute. Similarly to how security is improved by ensuring it is not breached, we propose that privacy can be warranted by ensuring it is not violated. As long as security is not breached, it is upheld. As long as privacy is not violated, it is protected.

By adopting the ideas behind the Privacy Violation Diagram as characteristics, a variant of the default quality model would look as follows:

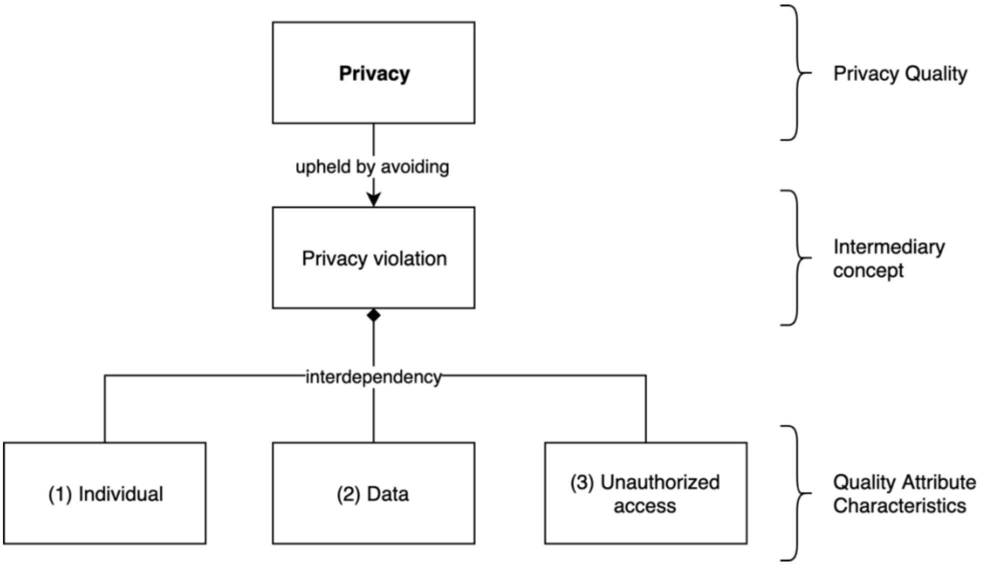


Figure 5: Privacy quality model, with the interdependent privacy characteristics of the Individual, Data and Unauthorized access.

In the model above, we have adapted the ISO/IEC 25010 specification of a quality model [62]. The quality model standard does not by itself support the interdependency concept, therefore, we have used the UML composition connector. The black diamond connecting the characteristics to the privacy violation concept indicates that the privacy violation is a composition that cannot exist without all of its components. We have also added the privacy violation as an intermediary concept to be able to address privacy.

As can be seen in Figure 5, the three characteristics we propose really are the characteristics of a privacy violation, rather than the characteristics of privacy itself. However, the goal of defining the characteristics of a quality attribute is to be able to address these and thereby improve the quality attribute. We reason that by addressing the characteristics of a privacy violation by applying architectural tactics, we do improve the overall degree of privacy in a system, thereby reaching our goal. Therefore, we will from here on refer to the characteristics of the privacy violation as “privacy characteristics”.

4.4 Privacy tactics

To address the proposed privacy characteristics, we have grouped and categorised several tactics derived from six selected privacy engineering studies. In the first section, we describe the method of how studies and tactics are selected and analysed. In the second section, we propose tactic groups that will be used to categorise the tactics. In section four, an analysis of the tactics of all six privacy engineering studies is presented. Section five contains a selection of tactics we propose to use to address privacy.

4.4.1 Method

To select valid privacy tactics, six privacy engineering studies were analysed. For analysis, the following steps were followed:

1. Selecting studies

By applying the study selection criteria from the systematic literature review in section 1.3.2, six studies containing privacy tactics were selected. The review protocol is as follows.

Goal

The goal of the systematic literature review is to gather relevant privacy conceptualizations. What “relevant” means depends on what the gathered information is used for. These criteria are formulated as study selection criteria in the next section.

Study selection criteria

As mentioned in section 1.3.2, to select studies worthy to examine, the following selection criteria were applied:

1. **Recent.** Only studies published after 2012 are included because of the impact of the GDPR, of which a first draft was published early 2012 [17].
2. **Applicable.** Only studies that are aimed towards conceptualizations of privacy used in software engineering methods or privacy engineering are included. Most studies include both.
3. **Topical**. Only studies that present their most recent privacy conceptualizations or privacy engineering methods are included. If conceptualizations or methods extend previous comparable studies, only the most recent study is included.

Search strategy

Software engineering information sources such as IEEE, SpringerLink, SEI and ACM were accessed through Google Scholar by submitting keywords. Comprehensive keywords such as "software architecture privacy", "software design privacy", "privacy engineering", "privacy by design" and "privacy conceptualizations" were combined to gather an initial set of studies.

Study selection procedures

The study selection criteria were applied by reading the abstracts of the studies and quickly scanning the studies' introductions for the keywords "definition", "concept", "conceptualization", "property", and "tactic", in combination with the word "privacy". The studies that fit the criteria were set apart for further examination.

Study quality assessment procedures

Due to the limited amount of available studies that fit the selection criteria, no separate quality assessment procedures were applied besides excluding pre-prints; studies that have not been published in a scientific journal and usually have not been peer-reviewed.

Data extraction strategy

After selecting the studies based on the selection criteria, the studies were read through completely. Basic information was extracted, summarizing information such as the authors, publication date, method name, and the name of the used privacy conceptualization. Examples are the GDPR, FIP or ISO 29100. Additionally, a short summary of the conceptualization was created. No special data extraction strategies were applied besides this.

Extracted data

The following studies were selected:

1. LINDDUN: A Privacy Threat Analysis Framework. Wuyts and Joosen, KU Leuven, 2014. [61]
2. NIST Privacy Framework. Hiller and Russell, NIST, 2017. [42]
3. A Critical Analysis of Privacy Design Strategies. Colesky et al., Radboud University, 2016. [11]
4. PRIPARE Methodology Handbook. Notario et al, Industry + Universities, EU and UK, 2015. [41]
5. Protection Goals for Privacy Engineering. Hansen et al, ULD Kiel Germany, 2015. [32]
6. PEAR: Privacy Enhancing ARchitectures. Kung, Trialog France, 2014. [40]

2. Analysing studies

To analyse the studies, we apply the grounded theory approach described in section 1.3.3. We provide the following information for each study:

- **Description**. We provide a general introductory description of the study.
- **Method**. If the study contains one or more method steps to follow, we describe how the method works and how it improves privacy. In some cases, we provide a Process-Deliverable Diagram to further illustrate how the method works and what data it produces or requires [58].
- **Architectural tactic suitability.** Some studies contain multiple conceptualizations of privacy or propose sets of strategies, tactics and patterns that work on different levels of detail, as described in section 2.5. This presents us with the question to what extent we can treat these conceptualizations as architectural tactics. We therefore describe for each method its architectural suitability and address potential caveats.
- **Tactic group creation.** We list the architectural tactics that the study contains, and by applying grounded theory create groups of tactics and describe them. The tactic

groups can be found in section 4.4.2. If we deem a tactic not fit for use because it does not address a privacy characteristic or because it does not fit the definition of architectural tactic, we remove the tactic and provide a rationale.

- **Visualization.** We map the tactics on the Privacy Violation Diagram to visualize the method. The mapping provides a quick and intuitive overview of how each method addresses the different characteristics of privacy.
- **Conclusion**. We summarize our findings. We focus on describing the completeness of the method by evaluating if and to what extent the tactics address privacy.

3. Tactic selection

After the analysis of each study and the creation of tactic groups, we create a selection of tactics that the architect may use to address privacy. The result of this can be found in section 4.4.4.

4. Privacy quality model

We extend the privacy quality model (Figure 5) with the proposed tactic groups, thereby creating a version of the privacy quality model that is as complete and usable for the software architect as possible.

4.4.2 Privacy tactic groups

In this section, we propose six main tactic groups that address privacy characteristics. The name of each tactic group is colour-coded and is used to categorise the privacy tactics of each study in their Privacy Violation Diagrams in section 4.4.2.2. For each tactic group, we also provide a short text indicating which tactics are placed in this group and why. A more elaborate explanation of our categorisation is provided at the corresponding analysis of each study, also in section 4.4.2.2.

The tactic groups are as follows:

1. Data awareness (characteristic: individual). Tactics in this group address the privacy characteristic of the individual. These tactics are aimed at raising the awareness of the individual to which data about him is processed. By raising awareness, the individual might reconsider with which data processors the individual shares its data, possibly leading the individual to choosing a data processor with a different approach to privacy. It might also lead to the wish for more granular data control by the individual, so the individual can choose which data it shares.

Data awareness tactics

The methods LINDDUN, Privacy Strategies & Tactics and PRIPARE [11, 41, 61] all provide a tactic in this category. The tactics in these studies are reasonably alike: they state that the individual should be informed about which data is processed about it and by whom.

2. Data control (characteristic: individual). Tactics in this group also address the first privacy characteristic of the individual. They aim to provide the individual with a more granular control over how a data processor handles its data, rather than forcing the individual to choose between using the services of data processor or none at all. One might argue that this group addresses the data characteristic rather than the individual, due to its name. However, because the tactics in this group focus on handing control to the individual, we consider these tactics that address primarily the individual.

Data control tactics

The studies Privacy Strategies & Tactics, Protection Goals and PRIPARE [11, 32, 41] provide tactics in this category. These tactics all are formulated largely the same; they focus on providing the individual with the option to give and withdraw consent and to provide the individual with the possibility to amend its data once it is being processed by a data processor.

3. Unauthorized access risk mitigation (characteristic: unauthorized access). This group contains tactics that aim to ensure that the privacy violation cannot be carried out at all, by applying tactics that aim to ensure no unauthorized access can take place in the first place. If data is secured properly, for example by applying information security principles or by tactics such as confidentiality and undetectability, the risk of a privacy violation occurring can be mitigated. Tactics in this group apply to the system as a whole, rather than just the data in it.

Violation risk mitigation tactics

Risk mitigation tactics are present in all privacy engineering methods in this analysis [11, 32, 41, 42, 61], except for PEAR [40]. A common tactic is confidentiality, which is present in the LINDDUN [61], NIST [34] and Protection Goals [32] studies.

4. Unauthorized access detection (characteristic: data). Tactics in this group actively contribute to detecting unauthorized accesses, thereby aiming to mitigate the impact of the unauthorized access. One could therefore also argue that tactics in this group are better suited as part of the unauthorized access mitigation group. However, because the tactics in that group are specifically aimed towards modifying the way data is stored in the system, we propose this as a separate group.

Tactics such as logging of modifications on data and auditing of reports are usually attributed to accountability tactics, but can also contribute to discovering whether unauthorized accesses have occurred. By discovering unauthorized accesses, possibly privacy violations, similar events in the future might be prevented. Also, an unauthorized access that is discovered earlier might have less impact because the data controller may be able to warn the subjects whose data was stolen.

Unauthorized access detection tactics

Few conceptualizations include tactics of this group. It is partially defined in PEAR as accountability; the logging of data and protection of logging to make it unforgeable [40]. In Protection Goals, we find the transparency tactic that specifies that the following: "The property that all privacy-relevant data processing – including the legal, technical and organizational setting – can be understood and reconstructed at any time." [32]. We reckon this tactic may be contribute to accountability and data breach discovery as well. However, we find most studies lack tactics that are aimed specifically aimed at unauthorized access detection.

5. Unauthorized access impact mitigation (characteristic: data). Tactics in this group address the second privacy characteristic. These tactics aim to mitigate the impact of unauthorized access, thereby preventing a privacy violation, by modifying the data aspect of a system. For example, by making sure that as little identifiable data as possible is available in the system, the chances of a privacy violation occurring in case of unauthorized access have become smaller.

Unauthorized access impact mitigation tactics

Tactics that mitigate the impact of unauthorized access, such as unlinkability and data minimalization, are present in all privacy engineering methods in this analysis.

Unlinkability and comparable variants are found in LINDDUN, NIST, Privacy Strategies & Tactics, PRIPARE and Protection Goals [11, 32, 41, 42, 61]. When unlinkability is practiced, the link between two items of interest (individuals, messages, data about the individual) is removed, meaning that it is more difficult to determine the connection between these items of interest in case of unauthorized access [46]. When less data is known about an identified individual in unauthorized access, the impact of the privacy violation is also mitigated. When unlinkability is practiced to such an extent that the individual cannot be identified in the breach, the privacy violation remains a 'mere' unauthorized access instead of a privacy violation. It can thus be argued that anonymization and pseudonymization are ways of achieving unlinkability.

Data minimization tactics are found in LINDDUN. If data minimization is practiced properly, the amount of data that is involved in the breach is reduced. Therefore, the impact of the unauthorized access can be mitigated. As noted by Hansen et al. [32], data minimization and unlinkability are related, because the removal of 'linking' data by applying unlinkability tactics can be regarded as a way of data minimization as well.

6. Consent compliance (characteristic: unauthorized access). Tactics in this group aim to ensure that any access to data is done with consent of the individual. We exclude organisational policies and procedures, because while they may help with ensuring that data is processed legally and with consent, they address internal organisational processes and not the software architecture design. In practice, this means we include tactics that ask the individual for consent, for example by asking for consent to place cookies on the individuals device. It also means we exclude NIST tactics such as "The workforce is informed and trained on its roles and responsibilities" [42].

4.4.2.1 Privacy tactic groups mapped on the Privacy Violation Diagram

The Privacy Violation Diagram below visualizes how the different tactic groups address the three proposed privacy characteristics. They are numbered according to the order in which they show their relevancy. This is further expanded upon in the next section, where we map the privacy tactic groups on a timeline.

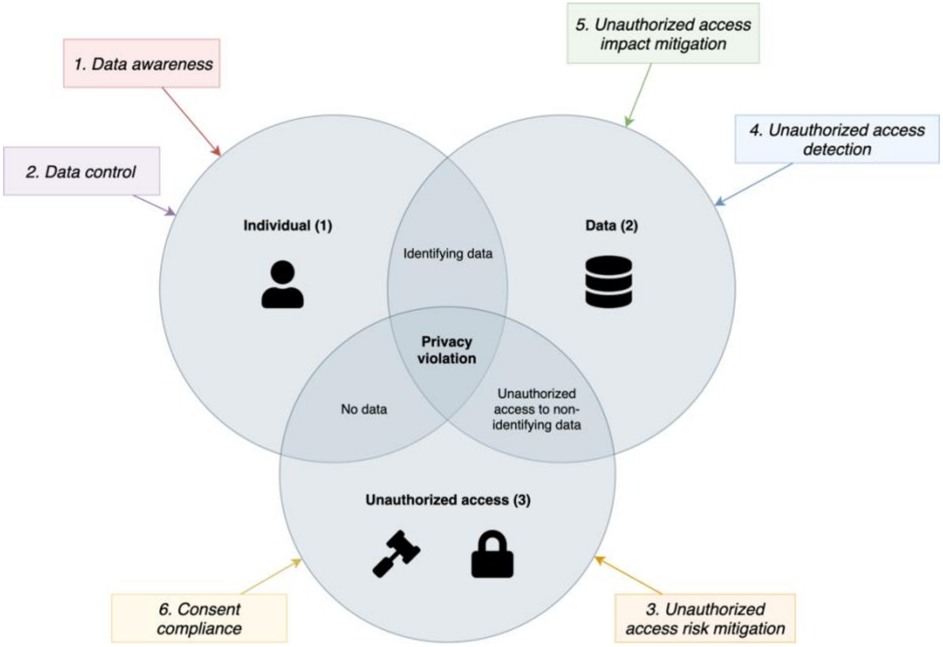

1. Data awareness
2. Data control
5. Unauthorized access impact mitigation
4. Unauthorized access detection
Individual (1)
Identifying data
Data (2)
Privacy violation
No data
Unauthorized access to non-identifying data
Unauthorized access (3)
6. Consent compliance
3. Unauthorized access risk mitigation

Figure 6: Architectural tactic groups for privacy mapped on the Privacy Violation Diagram

4.4.2.2 Privacy violation timeline

Earlier, we have proposed privacy as the interdependency between the individual, data and unauthorized access. However, dependency is not the only lens through which we can look at these characteristics. When perceived through the lens of time, we see that these characteristics need to occur in a specific order. We illustrate this and more in the privacy violation timeline in Figure 7. First however, we textually describe the sequence of the privacy violation diagram and the privacy tactic groups. We then relate both to the events on the privacy violation timeline.

Privacy violation diagram sequence

We here describe the sequence of the privacy violation diagram. As argued in section 3.5, for a privacy violation to happen, the presence of an individual is needed first. Without an individual, there is no one whose privacy can be violated, it thus needs to come first. Next, the data characteristic becomes relevant, because without data, the individual cannot be described. The individual can exist, but without data, its privacy cannot be violated (as noted in section 3.2, this study focusses on information privacy). This is also elaborated on in section 3.5. Unauthorized access as third and last characteristic can only occur as a last step, because for this, the characteristic of data needs to exist first. In Figure 7, this sequence is shown on the timeline.

Tactic groups sequence

Like the privacy characteristics, the tactic groups can too be seen as a sequence. This sequence describes the moments towards a privacy violation when these tactic groups become relevant to mitigate a privacy violation. An example would be the order of unauthorized access risk mitigation vs. unauthorized access impact mitigation, as can be seen in Figure 7. Unauthorized access risk mitigation has to happen before unauthorized access impact mitigation, because if data is never accessed without authorization, it also is not necessary to mitigate the impact of the access. We use parentheses to indicate which privacy characteristic becomes relevant at that particular step in the process.

The first tactic group, data awareness, becomes relevant before any data is in the system, for example if the individual (1) chooses not to entrust its data to the system. The risk of a privacy violation can hypothetically be ended here. However, we assume the individual will want to use the system. If the individual chooses to entrust its data to the system, the consent compliance group is applicable next. Without consent, any data processing done is inherently unauthorized, be it by the processor or a third party. For the next two tactic groups, data control and unauthorized access risk mitigation, we assume that the individual has chosen to use the system; its data is now in the system (2). The individual can choose to exercise a certain amount of control over it via data control tactics. For the owner of the system, it is advisable to implement tactics that protect the system against unauthorized access by applying access risk mitigation tactics.

For the next step in this sequence, unauthorized access (3) has been gained to the system. The next two tactic groups thus become relevant: unauthorized access detection and unauthorized access impact mitigation. A privacy violation can still be averted if impact mitigation tactics have been applied correctly, for example, by applying the unlinkability tactic. Lastly, a privacy violation can occur if all these tactic groups have failed to properly address the issue. This is indicated by the overlap of all three privacy violation characteristics.

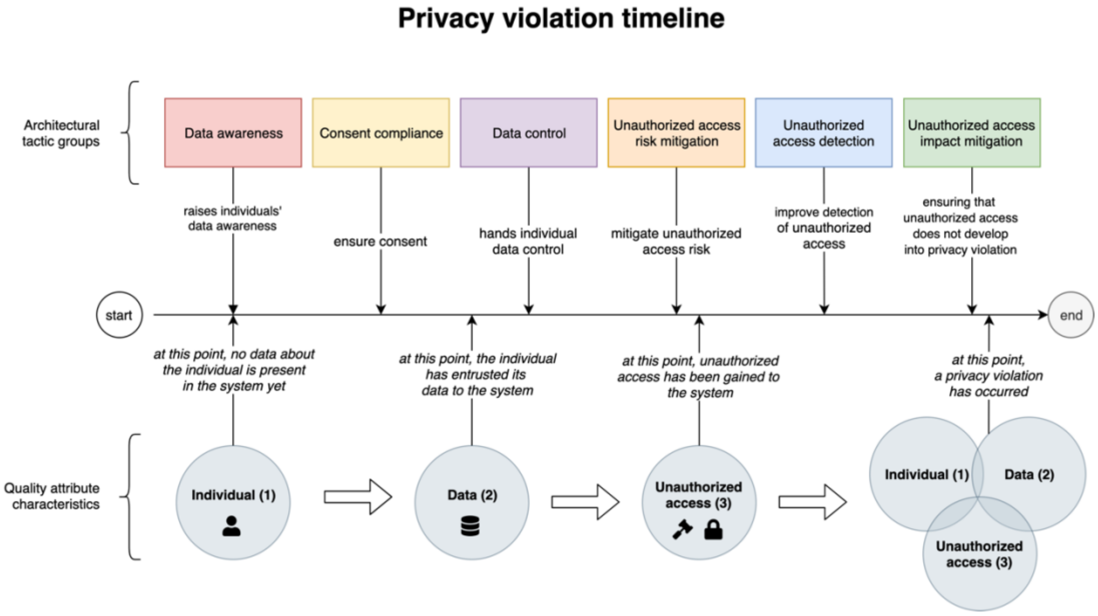

Privacy violation timeline
Architectural tactic groups
Data awareness
Consent compliance
Data control
Unauthorized access risk mitigation
Unauthorized access detection
Unauthorized access impact mitigation
raises individuals' data awareness
ensure consent
hands individual data control
mitigate unauthorized access risk
improve detection of unauthorized access
ensuring that unauthorized access does not develop into privacy violation
start
end
at this point, no data about the individual is present in the system yet
at this point, the individual has entrusted its data to the system
at this point, unauthorized access has been gained to the system
at this point, a privacy violation has occurred
Quality attribute characteristics
Individual (1)
Data (2)
Unauthorized access (3)
Individual (1)
Data (2)
Unauthorized access (3)

Figure 7: Privacy violation timeline including privacy characteristics sequence and tactic groups sequence.

The significance of Figure 7 lies in the clear distinction we make between privacy characteristics and architectural tactics. To create an architectural perspective, it is necessary to understand the differences between these two concepts. Furthermore, we think this overview may be suitable for architects and other stakeholders to support discussions about privacy.

4.4.3 Analysing studies

In this section, we present an analysis of privacy engineering studies. The goal of this section is to gather and categorize architectural tactics that improve privacy.

4.4.3.1 LINDDUN: a privacy threat analysis framework

Description

LINDDUN is a privacy threat modelling framework introduced in 2014. The goal of this framework is to improve privacy by determining and addressing privacy threats in a software architecture. Each letter of the LINDDUN method indicates a specific privacy threat type, such as Linkability.

In the LINDDUN study, privacy is not conceptualized. Rather the authors point the reader to the work of Solove [52] for a complete overview of the concept. The authors have however defined properties of privacy based on the work of Pfitzmann et al. [46], a study which distinguishes between hard privacy and soft privacy. The difference between the two, according to Pfitzmann et al., is that hard privacy means that as little data should be shared with third parties as possible, because third parties cannot be trusted due to lack of insight [61]. Soft privacy on the other hand assumes that data about the user has already been shared with third parties, and thus emphasizes techniques such as explicit consent and purpose to provide data protection.

Method

The privacy threats that are applicable in a specific system are discovered by determining the data flows, data stores, data processes and external entities of the system. It is suggested to do this by modelling the system as a Data Flow Diagram (DFD). For each data flow, store, process or external entity, the LINDDUN threat types are then applied, somewhat like how a software architect may apply perspectives to improve views. This results in a mapping of which the tactics are used to create so-called misuse cases. It is suggested to then use these misuse cases to create requirements that address them, which can be fulfilled by selecting and applying Privacy-Enhancing Technologies (PETs). A visualization of this method is provided as PDD in appendix section 7.2.1.

Architectural tactic suitability

We find that the LINDDUN privacy properties are well suited to use as architectural tactics, because they provide advice on how to address general issues with regards to privacy. They thus adhere to the definition we use. For example, like Pfitzmann [46], the LINDDUN authors describe addressing the issue of linkability between items of interest by applying unlinkability.

Categorized tactics

We here classify the architectural tactics of the LINDDUN study into tactic groups. An argumentation for each classification is provided.

1. Data awareness

- **Content awareness**. Content awareness is defined as "A property to make sure that users are aware of their personal data and that only the minimum necessary information should be sought and used to allow for the performance of the function to which it relates." [61]. According to the LINDDUN authors, content awareness is generally not explicitly defined in literature and therefore newly proposed.

  The authors describe that through increased content awareness, the user should better be able to decide which information to share, thereby probably sharing less data or with another party. One could therefore also argue that this tactic is part of the data minimization tactic group, and therefore a violation impact mitigation tactic. However, because this tactic is targeted at the individuals behaviour instead of data, it has been placed in the data awareness group.

2. Data control

LINDDUN does not contain tactics to improve data control for the individual.

3. Unauthorized access risk mitigation.

- **Plausible deniability**. "Plausible deniability refers to the ability to deny having performed an action that other parties can neither confirm nor contradict." [61]. This definition is newly proposed by the LINDDUN authors. We regard this tactic as unauthorized access risk mitigation tactic, because the tactic aims to ensure that data (which is how the action is expressed) does not reach the wrong party. For example, when user A sends a message to user B, we do not want user C to have any data on whether this action has happened or not.
- **Undetectability**. "Undetectability of an item of interest (IOI) from an attackers perspective means that the attacker cannot sufficiently distinguish whether it exists or not." [46]. The authors additionally state that in this tactic, the Item of Interest (IOI) itself is protected, rather than the relation of the IOI to its subject like with

anonymity or unlinkability. In the Privacy Violation Diagram, we see this as the data aspect where it does not overlap with the individual. This tactic contributes to mitigating the risk of unauthorized access, by hiding items of interest from a potential attacker.

- **Confidentiality**. In LINDDUN, confidentiality is defined as the "hiding of the data content or controlled release of data content." [61]. Hiding the content thus contributes to unauthorized access risk mitigation.

4. Unauthorized access detection

LINDDUN does not contain tactics aimed towards discovering unauthorized access.

5. Unauthorized access impact mitigation

- **Unlinkability**. This tactic is included in both the LINDDUN and Protection Goals studies, but defined somewhat differently. In LINDDUN, the definition by Pfitzmann and Hansen is used: "Unlinkability of two or more items of interest (IOIs, e.g., subjects, messages, actions, ...) from an attackers perspective means that within the system (comprising these and possibly other items), the attacker cannot sufficiently distinguish whether these IOIs are related or not." [46]. In case of unauthorized access, this may mean the attacker is less able to distinguish individuals or link data to individuals, thereby mitigating the impact of unauthorized access to data.
- **Pseudonymity**. Again, the definition by Pfitzmann and Hansen is used: "A pseudonym is an identifier of a subject other than one of the subjects real names. Pseudonymity is the use of pseudonyms as identifiers. A subject is pseudonymous if a pseudonym is used as identifier instead of one of its real names." [46]. As Colesky and Hoepman note, pseudonymity can also be perceived with respect to linkability [61]. We regard pseudonymity as a way to contribute to a degree of unlinkability, because by removing the link between the data and the individual and replacing it by a pseudonym, a degree of unlinkability is achieved.
- **Anonymisation** (LINDDUN). The definition by Pfitzmann and Hansen is used: "Anonymity of a subject from an attackers perspective means that the attacker cannot sufficiently identify the subject within a set of subjects, the anonymity set." [46]. Like with pseudonymization, we regard anonymisation as a way to achieve unlinkability.

6. Consent compliance

- **Policy & consent compliance** (LINDDUN). This tactic requires the data controller to "inform the subject about the system's privacy policy and allow the subject to specify consents in compliance with legislation before accessing the system" [61]. It aims to ensure the implementation and enforcement of system policy compliancy and user consent.

Privacy Violation Diagram

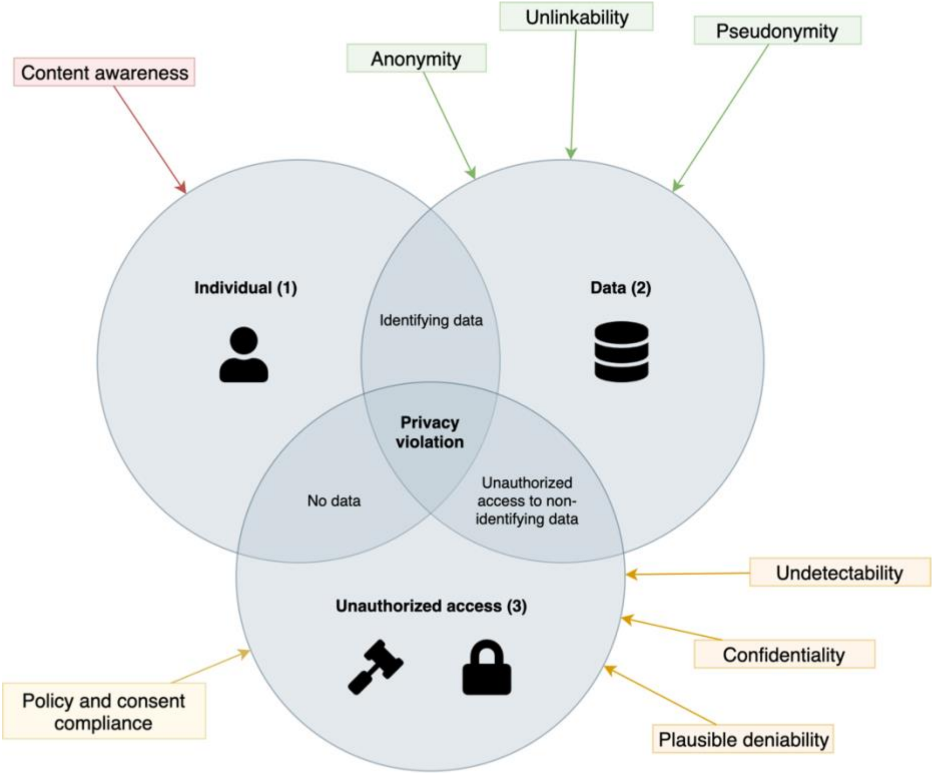


Figure 8: LINDDUN in the Privacy Violation Diagram

Conclusion

The LINDDUN architectural tactics mainly improve privacy by addressing the data characteristic of privacy. It does however not contain tactics that are aimed towards unauthorized access detection. Also, while LINDDUN provides a data awareness method for the individual, it does not take the next step of providing more data control.

4.4.3.2 NIST Privacy Framework

Description

The NIST Privacy Framework was introduced in 2017 the U.S. National Institute of Standards and Technology [34]. The main goal of the framework is to improve privacy in organisations as a whole through several methods, summarized in the Enterprise Risk Management framework. The framework contains two key components, elaborated on in the method section. Enterprise Risk Management focuses not just on improving privacy through software engineering practices, but supports organisations with managing privacy risks at an organisational level as well [42].

Method

The NIST Privacy Enterprise Risk Management Framework features two large key components; the Privacy Framework, and the Privacy Risk Management approach.

The first component, the Privacy Framework, consists of three parts: the Core, Profiles , and Implementation Tiers. The Core (1) is a set of privacy protection activities to support communication about privacy within and between companies. Profiles (2) are collections of privacy activities or outcomes of an organisation. They can be used to measure the current state of privacy in an organisation. Lastly, Implementation Tiers (3) indicate levels of how organisations view privacy risks. Implementation Tiers are meant to support organisations in improving their privacy practices by creating concrete goals to strive to.

The second component is the Privacy Risk Management approach, based on the NIST Security Framework. The risk management approach provides us with a method to calculate privacy by multiplying the likelihood of a problematic data action with the impact of the problematic data action. A visualization of this method is provided as PDD in appendix section 7.2.2.

Architectural tactic suitability

The NIST Privacy Framework presents two main challenges: its size, and the fact that the framework is quite a bit broader in scope; it focusses on the whole organisation rather than just its used software systems. The framework specifies five function groups, each consisting of several categories with several subcategories of activities to improve privacy in an organisation. The activities can be regarded as architectural tactics. However, most of those activities focus on organisational policies. We have therefore chosen to only describe the included activities and not to describe the excluded activities. A full description of all excluded tactics would not contribute to a more clear and concise selection of tactics.

Categorized tactics

In this section, the architectural tactics of the NIST framework are classified into tactic groups. An argumentation for each classification is provided. NIST chooses to name the tactics by specifying their function group, function and category. For example: CT.DM-P1 means Group: Control, Function Group: Data Processing Management, Function: Privacy 1. We have chosen to also give each activity a name that is more easily readable.

1. Data awareness

- **CT.DM-P1: Data review.** This tactic describes that data elements can be accessed for review, presumably by the individual who the data is about. It thus contributes to data awareness.
- **CT.DM-P2: Data disclosure.** Data disclosure tactic is aims to ensure that data elements can be accessed for transmission or disclosure. Presumably, that means that the data processing organisation needs to have the technical ability to disclose information to the individual. We therefore consider this a tactic that contributes to data awareness.
- **CM.AW-P1: Purpose communication.** This tactic specifies “mechanisms” to communicate data processing purposes, practices and privacy risks. We can interpret mechanisms as both organisational procedure and architectural tactic, so we choose to interpret it as tactic. By informing the individual for what purpose its data is processed, this tactic contributes to data awareness.
- **CM.AW-P3: Visibility.** By making data processing “visible”, the individual (or data processor) should have more insight and thus awareness about the processed data. We deem this tactic largely comparable to the data review, data disclosure and purpose communication tactics.
- **CM.AW-P5: Data correction communication.** By communicating and corrections or deletions of data to the individual, the individual is more aware of which of its data is processed.

- **CM.AW-P7: Breach notification.** By notifying the individual of any data breaches, the individual can be made aware of unauthorized access to data and possible privacy violation.

2. Data control

- **CT.DM-P3: Data alteration.** The data alteration tactic describes that data elements can be accessed for alteration. If this can be done by the individual, this contributes to data control. However, it is not specified clearly if this is to be done by the individual or the data processor. We interpret it as by the individual.
- **CT.DM-P4: Data deletion.** By providing ways to delete its data, the individual is given more data control.
- **CM.AW-P8: Individual mitigation.** This tactic describes mitigation mechanisms, such as credit monitoring, consent withdrawal, data alteration and deletion. It therefore is a mix of the two data control tactics mentioned earlier in this group, combined with consent. We therefore choose to place this tactic both in the data control and consent compliance group.

3. Unauthorized access risk mitigation.

- **PR.AC-P1: Credentials.** This tactic specifies to issue, manage, verify, revoke and audit authorized individuals, processes and devices. By ensuring only authenticated individuals can access data, the risk of unauthorized access can be mitigated.
- **PR.AC-P4: Authorization.** The authorization tactic aims to ensure access permissions and authorizations to incorporate the principles of least privilege and separation of duties. This ensures logged in individuals only have access to the resources they are entitled to, thereby contributing to a lowered risk of unauthorized access.
- **PR.AC-P5: Network segregation.** By segregating parts of the network used by the system, if one part of the network is compromised, another part may stay secure.
- **PR.DS-P1: Data-at-rest protection.** This tactic is quite generic. NIST does not specify it any further than "Data-at-rest are protected". Yet, we have chosen to include the tactic, because any contribution to data protection may lower the risk of unauthorized access to it.

- **PR.DS-P2: Data-in-transit protection.** Like with data-at-rest protection, data-in-transit is not defined further than “data-in-transit are protected”. For the same reasons as data-at-rest protection, we include this tactic.

4. Unauthorized access detection.

- **CM.AW-P4: Log data disclosures.** Logging data disclosures and sharing of data for review may contribute to discovering any unauthorized access to it. It therefore is categorized in the unauthorized access detection group.
- **CM.AW-P6: Maintain data provenance.** The data provenance and lineage tactic aims to be able to reconstruct modifications to data. This contributes to integrity, but may also contribute to tracing back unauthorized access to the data.

5. Unauthorized access impact mitigation.

- **CT.DP-P1: Limit observability and linkability.** This tactic proposes to limit observability and linkability of data, for example by applying cryptography or keeping data on local devices. These techniques thereby contribute to a lower impact if any data is accessed illegally.
- **CT.DP-P2: Limit identification of individual.** By applying techniques that limit identification, such as tokenization, the impact of unauthorized access to data can be mitigated.
- **CT.DP-P3: Limit inferences.** This tactic proposes techniques such as decentralized data processing and a distributed architecture to limit inferences if unauthorized access to the data was obtained.
- **CT.DP-P4: Selective collection.** By configuring user devices in such a way that data is selectively collected or disclosed, the amount of data that ends up in the system can be limited.
- **CT.DP-P5: Attribute substitution.** By substituting attribute values by attribute references whenever data is shared, the accuracy of the data can be lessened while it keeps its meaning. For example, the age attribute “older than 18” is less accurate than an exact birthdate, but may convey the same meaning and may satisfy the requirement of knowing if someone is of age.

6. Consent compliance

- **CM.AW-P8: Individual mitigation.** This tactic describes mitigation mechanisms such as consent withdrawal. We therefore consider it a consent compliance tactic.

Privacy Violation Diagram

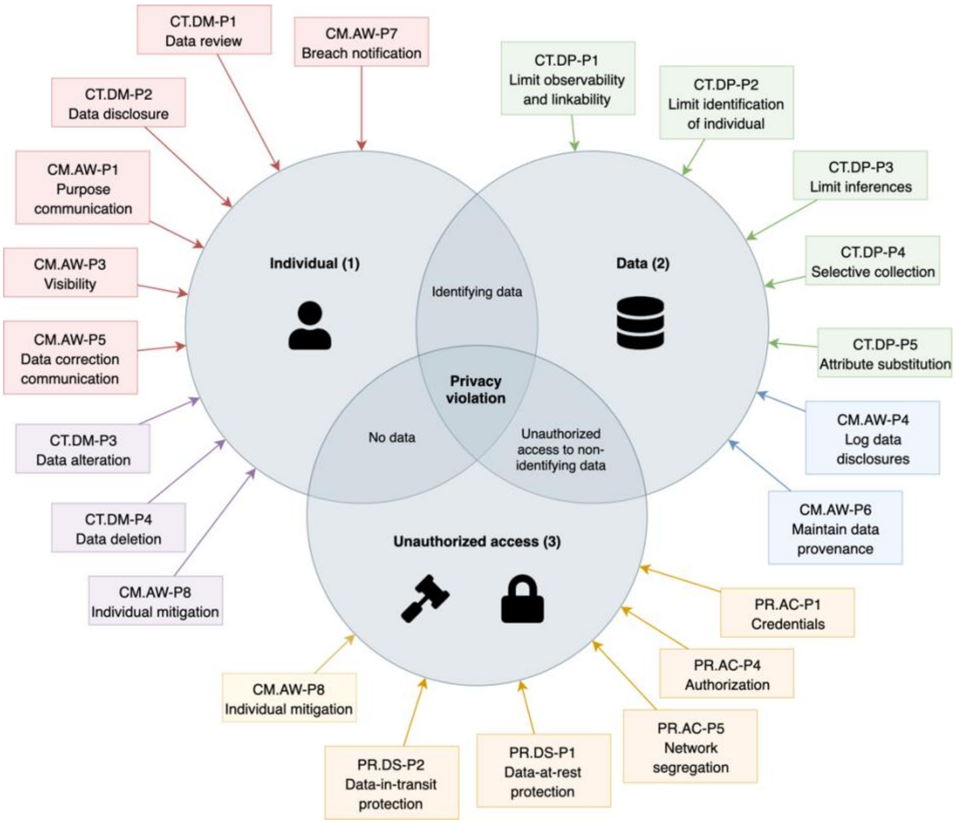


Figure 9: The NIST Privacy Engineering and Security Objectives in the Privacy Violation Diagram

Conclusion

The NIST tactics address all privacy characteristics at least once and therefore is one of the most complete methods in this study. However, most tactics are formulated not very concisely, causing them to leave room for interpretation to the architect. Despite this, the general purpose of each tactic is often clear. Compared to other methods, NIST contains a relatively large amount of data awareness tactics. A large part of the NIST activities was not included because they contain organisational policies rather than organisational tactics.

4.4.3.3 A Critical Analysis of Privacy Design Strategies

Description

Colesky et al. [11] explore the relationship between privacy requirements based on the GDPR, privacy strategies, tactics and patterns. Specifically, they propose the addition of the 'privacy tactics' level of abstraction between the levels privacy strategies and privacy patterns. They note that Privacy by Design has been proposed to reach the goal of translating privacy requirements into software. However, due to a lack of developer tools, they propose their new method based on privacy design patterns.

They state they regard privacy as a quality attribute, but prefer the term 'privacy protection' rather than 'privacy'. While they regard privacy protection as a quality attribute, characteristics of privacy have not been formulated. Instead, Colesky et al. map their strategies and tactics on certain events called actions, which address GDPR principles. The actions are called operate, store, retain, collect, share, change and breach and are used to address one or more ways of processing data as specified in the GDPR.

Method

An approach to apply the actions, strategies and tactics may look as follows. A visualization of the steps is provided as PDD in section 7.2.3.

1. Determine relevant actions. Depending on the type of the project, not all actions will apply. For example, if the organisation does not process sensitive data at a large scale, a data protection officer may not be needed [21]. It therefore is necessary to determine which actions are relevant for the project at hand.
2. Map privacy strategies on relevant actions. By selecting the privacy strategies that correspond with the appropriate actions, the privacy requirements that result from the GDPR can be addressed.
3. Amend or create privacy requirements by selecting suitable privacy tactics and patterns. By selecting the privacy tactics and patterns that correspond with the selected privacy strategies and transforming them into requirements, a new set of improved, more concise system requirements can be created.

Architectural tactic suitability

Colesky et al. note that architectural tactics exist and deem their privacy tactics compatible with the definition. They define a privacy tactic as “an approach to privacy by design which contributes to the goal of an overarching privacy design strategy.” [11].

Categorized tactics

In this section the architectural tactics of the Privacy Strategies and Tactics study are classified into tactic groups. An argumentation for each classification is provided.

1. Data awareness

These strategies are aimed at raising user awareness by informing them about how their data is processed. They are therefore placed in the data awareness tactic group.

- **Supply**. The supply tactic aims to inform the individual by making available resources that describe how personal data is processed and where potential risks are.
- **Notify**. The notify tactic then actively informs the individual about how and which data about it is processed.
- **Explain.** The explain tactic specifies that information about data processing should be made concise and understandable, for example by using privacy icons [11].

2. Data control

These data control tactics relate to the data subject’s control over their information’s collection, storage, operation and dissemination [29]. They are therefore placed in the data control tactic group. Colesky et al. comparatively call the collection of these tactics the ‘CONTROL’ strategy. The CONTROL strategy originally contains an extra tactic: ‘consent’. We have moved this tactic to the Consent compliance group though, because we think it addresses the unauthorized access characteristic rather than the data characteristic.

- **Choose.** The choose tactic allows the individual to choose up front what data to share.

- **Update**. The update tactic allows the individual to change its data once it is already in the system.
- **Retract.** The retract tactic allows the individual to remove its data that is already in the system.

3. Unauthorized access risk mitigation

- **Restrict**. The restrict tactic is defined as "preventing unauthorized access to personal data" [11]. We consider the restricting tactic part of the risk mitigation group, because this tactic is aimed at ensuring that no data is accessed illegally.
- **Distribute**. The distribute tactic is defined as "partitioning personal data so that more access is required to process it". By partitioning data in a system or spreading it over multiple systems and requiring more access, it is more difficult to access personal data in the first place. We thus consider this a risk mitigation tactic.
- **Isolate.** Defined as "processing parts of personal data independently, without access or correlation to related parts". This tactic closely resembles the distribute tactic and is thus considered a risk mitigation tactic.
- **Disassociate.** The dissociate tactic features delayed routing, or batched routing PETs [31]. By batch sending incoming packets, the correlation between these pieces of potentially personal data can be removed. This decreases the risk of being able to access the data.

4. Unauthorized access detection

This study proposes several tactics that can be used to discover unauthorized access to data, such as log and report. However, these tactics are aimed at achieving consent compliance rather than unauthorized access detection. They are therefore placed in the consent compliance group.

5. Unauthorized access impact mitigation.

We consider these tactics as part of the unauthorized access impact mitigation group, because they are aimed at softening the impact of a data leak after it has happened, by making re-identification more difficult. Depending on the situation at hand, one or more of these tactics can be combined to achieve more minimized and unlinked data in a system.

- **Exclude**. Excluding data means to refrain processing certain personal data of the individual. We note this is somewhat comparable to the choose tactic, the difference being that the exclude tactic is seen from the perspective of the data controller, while the choose tactic assumes the perspective of the data subject.
- **Select**. The select tactic is comparable with exclude tactics, the difference being that select tactics assume no personal data is shared in the first place and that any personal data sharing has to be selected specifically. Exclude tactics assume the other way round; it is chosen which data to exclude afterwards.
- **Strip**. The strip tactic proposes to remove unnecessary metadata from the individual in the system.
- **Destroy**. The destroy tactic aims to remove all personal data of the individual, for example after the lawful data retention time has passed.
- **Mix.** The mix tactic proposes to create mix networks that reduce correlation between different pieces of personal data.
- **Obfuscate.** Obfuscate tactics aim to apply encryption on personal data so it cannot be read if it is accessed illegally.
- **Summarize, Group.** By summarizing or grouping data, the data contains fewer details. It depends on the use case at hand whether it is still possible to accomplish the necessary tasks with grouped data.

6. Consent compliance

These tactics aim to facilitate acquiring user consent. They address the 'unauthorized' part in the 'unauthorized access' characteristic.

- **Consent.** Consent tactics determine that the individual should be given the ability to choose to consent to their data being processed or not.
- **Log, Report.** By generating logs and reports of how data is processed, auditing can take place which is sometimes necessary to demonstrate consent compliance.

Not architectural tactics

- **Create, Maintain, Uphold, Audit.** We choose to exclude these tactics because they specify organisational policies and procedures, rather than ways to improve the technical architecture of the system. For example, the create tactic specifies to "acknowledging the value of privacy and deciding upon policies which enable it, and processes which respect personal data" [11]. While useful, this tactic can only be realised by specifying organisational policies.

Privacy Violation Diagram

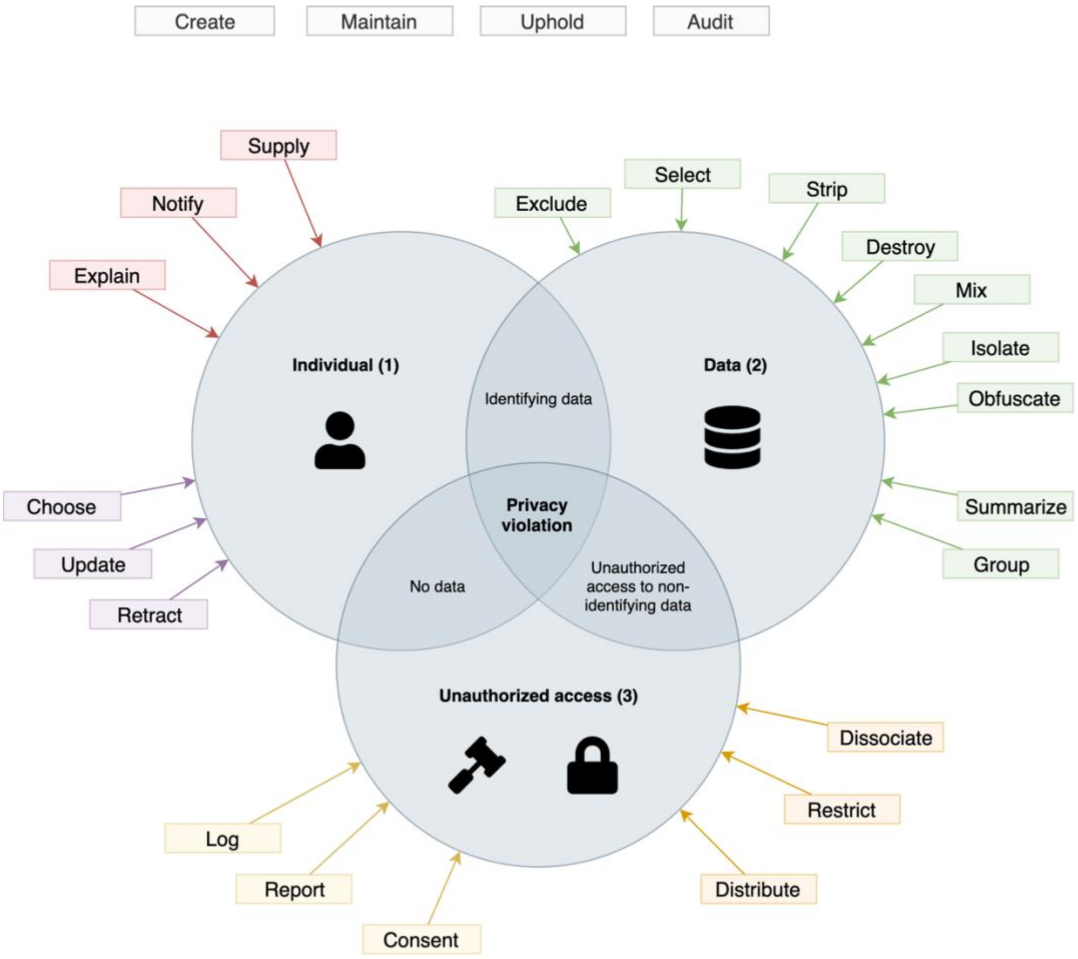


Figure 10: Privacy Strategies & Tactics in the Privacy Violation Diagram

Conclusion

The tactics proposed by Colesky et al. are quite extensive and address the privacy characteristics well. All tactic groups are represented with at least three tactics per group, making this one of the most complete privacy engineering studies in this research. We regard some of the tactics as organisational procedures to achieve GDPR compliancy rather than architectural tactics, causing them to fall outside of the scope of the PVD.

4.4.3.4 PRIPARE

Description

PRIPARE is a privacy-by-design methodology funded by the EU and published in 2013 [43]. The PRIPARE handbook describes two goals; to facilitate the application of a privacy-by-design and security-by-design methodology, and to foster a risk management culture by creating educational materials [41]. The PRIPARE project has released an extensive handbook on PbD practices, containing existing standards, practices and new research proposals on privacy engineering. We focus on the privacy-by-design methodology and PbD practices.

Method

PRIPARE is meant to be used as both a reference book and as a practical guide to improving privacy in an organisation. The PRIPARE methodology is structured in seven phases that "match common and classic system engineering phases" [41]. A somewhat abstract Process-Deliverable Diagram of PRIPARE can be found in the appendix in section 7.2.4. The phases of the methodology are as follows:

1. **Analysis**, when privacy and security issues and concerns are discovered so they can be addressed later. This is done by characterizing the system through any popular method, for example by drawing diagrams.
2. **Design**, which is then the system's architecture, components, modules and interfaces are determined to satisfy the stakeholders' requirements.
3. **Implementation**, which is when the system design is realized in working software.
4. **Verification**, used to ensure that the system meets its privacy and security requirements. Generally done through security and privacy testing, evaluation and audits.
5. **Release**, which is when the system is delivered to the customer. PRIPARE determines that the software producing organisation should have an action plan ready in case of any privacy and security issues.
6. **Maintenance**, which is a continuous phase once the system is in active use. The main goal is to make sure that privacy and security processes and rules are being enforced.
7. **Decommission**, which determines that systems should be dismantled correctly and that personal data should be treated according to legislation and policies.

Architectural tactic suitability

PRIPARE has adopted 11 privacy principles from the ISO 29100 standard. Privacy principles are defined as follows: "A privacy principle is both an essential consequence of privacy that defines its foundations and a feature that a system must compulsorily exhibit to respect the user's privacy" [41]. The authors have chosen to adopt the ISO standard rather than proposing their own set of principles based on the GDPR, because they regard the ISO 29100 standard as the worldwide leading privacy framework standard, and because it is a reference document aimed at privacy engineers rather than jurists. The ISO 29100 privacy principles can be regarded as architectural tactics; they provide general advice to improve privacy in a system. They can also quite easily be mapped on the privacy characteristics we propose.

Categorized tactics

In this section, the architectural tactics of the PRIPARE method are classified into tactic groups. An argumentation for each classification is provided.

1. Data awareness

- **Openness, transparency and notice**. "Providing PII principals information about how the PII is processed." [14]. This tactic specifies to provide information to individuals about the type of data processed, for what purpose and to whom the data is disclosed. It thereby contributes to data awareness.
- **Individual participation and access**. "Giving PII principals the ability to access and review their PII… -." [14]. This tactic is described more extensively and focusses on providing corrections to the PII as well. Here, we only cite the first part of the definition, because parts two, three and four of the definition focus on data corrections and amendments by the PII. We place data corrections and amendments in our separate data control tactic group, rather than the data awareness group.
- **Purpose legitimacy and specification**. This tactic specifies that the purpose of the processing of PII should be in line with applicable law [14]. We do not regard this an architectural tactic, but rather an organisational procedure. However, it also specifies that the purpose(s) of the collection of the PII should be communicated towards the data subject before any processing is done, and that the purpose should be explained as clearly as possible. We therefore categorize this as a data awareness tactic.

2. Data control

- **Individual participation and access**. This tactic is included both in the data awareness tactic group and the data control tactic group, because it specifies tactics that target both groups. This part of the tactic focusses on the "Allowing PII principals to … - … amend, correct or remove the PII." [14]. It thus specifies to hand control to the individual by allowing them to amend, correct and remove identifying information.

3. Unauthorized access risk mitigation

- **Information Security**. The information security tactic focusses on ensuring the confidentiality, integrity and availability of personally identifiable information, and protect it against unauthorized access, destruction, use, modification, disclosure or loss [14]. This tactic is aimed towards ensuring that information is kept safe. We thus consider this unauthorized access risk mitigation tactic.

4. Unauthorized access detection

PRIPARE does not contain tactics aimed towards unauthorized access detection.

5. Unauthorized access impact mitigation

- **Collection limitation.** This tactic is defined as "limiting the collection of PII to that which is within the bounds of applicable law and strictly necessary for the specified purpose(s)." [14]. By only collecting as little information as possible, the impact of unauthorized access to the data can be mitigated.
- **Data minimization**. The data minimization tactic is defined as the minimization of the processing of PII [14]. This is done by minimizing the number of people who have access to the PII to just those who strictly need it, by offering default options that do not require processing of PII, and by deleting PII when its purpose has expired.
- **Use, retention and disclosure limitation**. This tactic is defined as "limiting the use, retention and disclosure (including transfer) of PII to that which is necessary in order to fulfil specific, explicit and legitimate purposes." [14].

6. Consent compliance

- **Consent and Choice**. This tactic is formulated as “Presenting to the PII principal the choice whether or not to allow the processing of their PII.” [14]. Exceptions are situations where consent cannot be withheld or where the law specifically allows PII to be processed without consent.

No privacy architectural tactic

- **Accuracy and quality**. This tactic specifies that PII is kept accurate, complete and up-to-date [14]. We do not consider this a privacy tactic, but rather a tactic to ensure data quality. We reckon that out-of-date or incorrect information is in a system does not necessarily lead to a greater risk of privacy violation.
- **Privacy compliance**. Adhering to the tactic of privacy compliancy means that all data processing meets requirements with regards to data protection and privacy safeguarding by conducting audits, by having appropriate internal controls and independent supervision mechanisms to assure compliance with the law, and by developing and maintaining privacy risk assessments to check if the processing of PII is done in according with data protection and privacy requirements. [14]. We do not consider this an architectural tactic because it specifies organisational procedures to improve privacy, rather than ways of improving architecture design.
- **Accountability.** This tactic is aimed mainly at enforcing privacy policies within an organisation. Accountability is regarded as the “adoption of concrete and practical measures for protecting personally identifiable information” [14]. It also specifies “documenting and communicating […] all privacy-related policies, procedures and practices”. These are not architectural tactics, but are organisational policies.

Privacy Violation Diagram

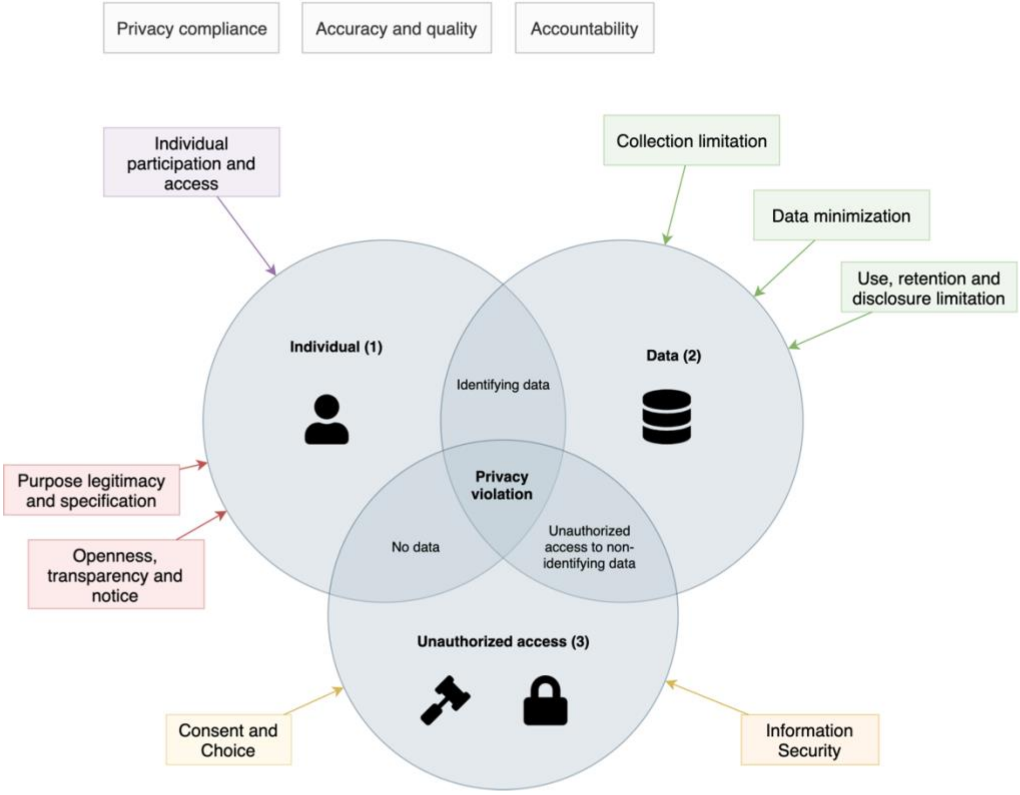


Figure 11: PRIPARE in the Privacy Violation Diagram

Conclusion

The PRIPARE tactics address each characteristic with at least two tactics and all tactic groups are represented at least once. The tactics are described on a somewhat higher level than usual for architectural tactics, potentially limiting their practical applicability. PRIPARE contains three tactics that are aimed at organisational policies which are therefore excluded. We regard the PRIPARE tactics as one of the most complete sets of tactics to address privacy.

4.4.3.5 Protection Goals for Privacy Engineering

Description

The Protection Goals for Privacy Engineering study was published by the German Independent Centre for Data Protection in Kiel in 2015 [32]. The study specifies six goals that provide privacy protection. Three of those are the often seen triad of CIA security protection goals: confidentiality, integrity and availability. Hansen et al. have then taken a comparable approach to the NIST and have added three extra privacy protection goals. Interestingly, these goals are not the same as proposed by the NIST. Whereas the NIST adds the goals of disassociability, predictability and maintainability, Hansen et al. add the goals unlinkability, intervenability and transparency. An important aspect of the study is the supposed oppositeness of some of the protection goals, which is why the goals are mapped along mirrored axes. For example, the protection goal of availability is mirrored with confidentiality, because by improving confidentiality, less information is available to some users, thereby decreasing overall availability of information to the user [32].

Method

There no specific method to apply the protection goals, although it is proposed that software engineers keep them in mind when building a system. It is also claimed that the protection goals are compatible with the Privacy by Design philosophy by Cavoukian.

Architectural tactic suitability

The privacy protection goals can be regarded as architectural tactics to a reasonable extent, because they do provide general advice to improve privacy in a system. At the same time, they are also formulated quite abstractly, making them somewhat more difficult to apply in practice. However, because they can be mapped on the privacy characteristics we propose, we regard them as architectural tactics.

Categorized tactics

In this section the architectural tactics of the Protection Goals study are classified into tactic groups. An argumentation for each classification is provided.

1. Data awareness

The Protection Goals do not contain tactics to specifically improve data awareness for the individual.

2. Data control

- **Intervenability**. The intervenability tactic is proposed as "the property that intervention is possible concerning all ongoing or planned privacy-relevant data processing" [32]. It is focussed on the individual whose data is processed and "reflects the individuals' right to rectification and erasure of data, the right to withdraw consent". We therefore deem this a data control tactic.

3. Unauthorized access risk mitigation

- **Confidentiality**. This security quality attribute property is often used in conjunction with integrity and availability, together forming the AIC-triad. In Protection Goals, it is defined as "the need for secrecy, i.e. the non-disclosure of certain information to certain entities within the IT system in consideration" [32].
- **Integrity**. Integrity expresses the need for reliability and non-repudiation regarding a given piece of information, i.e. the need for processing unmodified, authentic, and correct data. [32]. We consider this a violation risk mitigation tactic, because guaranteeing the reliability of the communicated data may reduce the risk of it being intercepted, thus reducing the risk of a privacy violation.

4. Unauthorized access detection

- **Transparency**. Transparency is defined as "The property that all privacy-relevant data processing – including the legal, technical and organizational setting – can be understood and reconstructed at any time." [32]. Hansen et al. describe that transparency can be achieved by, for example, logging and reporting. We regard this as both unauthorized access detection and a consent compliance tactic, because logging and reconstructing of data may help with both.

5. Unauthorized access impact mitigation

- **Unlinkability**. This tactic is included in both the LINDDUN and Protection Goals studies, but defined somewhat differently. In the Protection Goals study, unlinkability is defined as "the property that privacy-relevant data cannot be linked across domains that are constituted by a common purpose and context." [32]. Hansen et al. describe the link between unlinkability and data minimization; by minimizing data, there is less data available to be linked, thus lowering the chances and impact of any unauthorized access to the data.

6. Consent compliance.

- **Transparency**, see unauthorized access detection.

No architectural privacy tactic

- **Availability**, defined as follows: "Availability represents the need of data to be accessible, comprehensible, and processable in a timely fashion" [32]. We do not consider this an architectural privacy tactic, because a system that is unavailable does not necessarily compromise privacy.

Privacy Violation Diagram

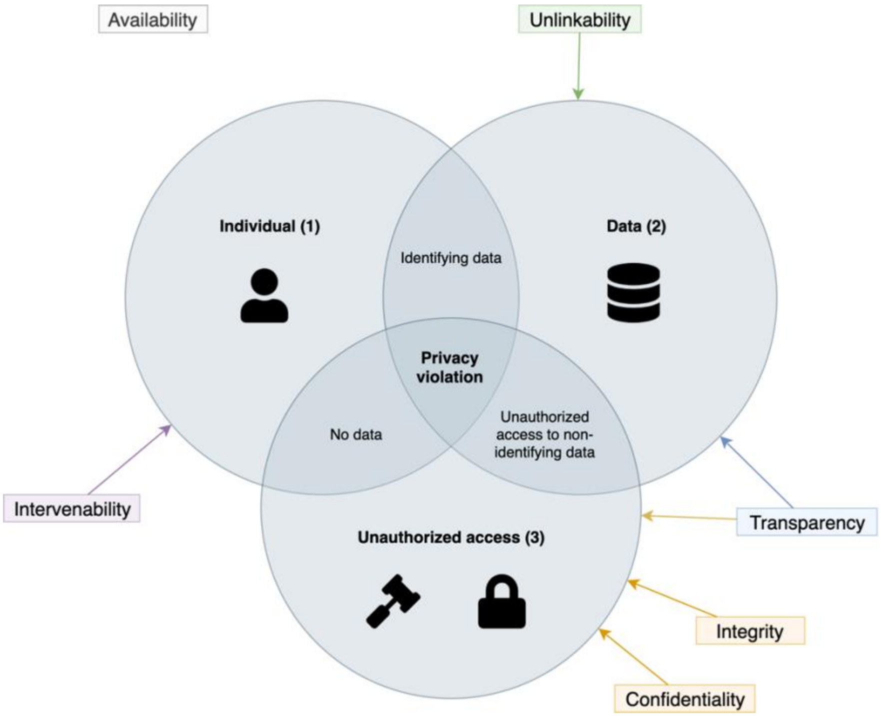


Figure 12: Six Protection Goals in the Privacy Violation Diagram

Conclusion

Hansen et al. provide six protection goals of which five contribute to mitigating a privacy violation, according to the privacy violation diagram. Despite that most protection goals are formulated as -ilities, we reckon they can be used as tactics because they aim to improve privacy without them being part of it. The protection goals are formulated quite broadly, which somewhat limits their practical applicability. All three privacy violation characteristics are addressed by at least one protection goal, although not all tactic groups are represented. No protection goals are specified that address the data awareness tactic group.

4.4.3.6 PEAR: Privacy Enhancing Architectures

Description

In the PEAR study, Kung describes the Software Engineering Institute software architecture process [3, 49] and relates this to privacy. A breakdown of the privacy quality attribute into characteristics is not provided, but several architectural tactics and tactic groups are proposed. It is not made clear where the tactics and tactic groups are derived from however. The tactic groups are textually described, but the tactics themselves are not. The interpretation of how the tactics have to be applied is therefore left up to the reader.

Method

Kung describes the application of architectural tactics, patterns and quality attribute scenarios to improve the privacy quality attribute in a system. It is stated that "in order to re-use, architecture tactics are often described through patterns" [40], but the *context, problem* and *solution* structure by Bass et al. [3] is not used. Kung describes several other methodologically relevant studies, such as the ATAM [36] requirements elicitation method and the CBAM cost-benefit analysis method. However, the relation of these methods to specifically privacy is not described; only a generic description of each method is provided and proposed to be used.

Architectural tactic suitability

The architectural tactics in this study are not textually described, but the tactic groups are. We thereby have to interpret the meaning of the tactics ourselves to some degree. Besides this, the tactics are based on the same principles of architectural tactics we use in this study, so they are reasonably suitable to analyse as architectural tactics as defined by Rozanski and Woods [49].

Categorized tactics

In this section the architectural tactics of the PEAR study are classified into tactic groups. An argumentation for each classification is provided.

1. Data awareness

PEAR does not provide data awareness tactics.

2. Data control

PEAR does not provide data control tactics.

3. Unauthorized access risk mitigation.

- **Change Crypto Strength and Method, Change Protection Strength**. In the PEAR study, these tactics are categorized in a tactic group called Modifiability, described as "tactics that are needed to cope with evolution needs". We do not consider the tactics as privacy-improving when explained as such, because we consider modifiability a separate quality that is part of the ISO 25010 maintainability group [62]. However, improving crypto strength and method and the data protection strength of a system will probably contribute to a lowered risk of unauthorized access to data. We therefore categorize these tactics in the unauthorized access risk mitigation group.

4. Unauthorized access detection

- **Log data transactions**, **Log modifications**, **Protect log data**. In the PEAR study, these tactics are categorized as part of the Accountability group. Accountability tactics are described as the "logging of relevant events [...] and protection of logging" [40]. As per the ISO 25010 standard however, we consider accountability a quality that contributes to security [62]. At the same time, extensive logging of events and modifications may contribute to the discovery of potential unauthorized accesses. We therefore choose to categorize these tactics in the unauthorized access detection group, although they may be categorized as consent compliance tactics as well, if logging of events is required by local privacy legislation.

5. Unauthorized access impact mitigation

- **Anonymize credentials**, **Limit processing**. In the PEAR study, these tactics are regarded part of the Minimization tactics group. We place these tactics in the unauthorized access impact mitigation group, the same group in which we place data minimalisation tactics, because the anonymize credentials and limit processing tactics contribute to less data in the system, thereby lowering the impact of a potential unauthorized access.

6. Consent compliance

PEAR does not contain architectural consent compliance tactics.

No architectural privacy tactic

- **Change policy.** We do not consider the change policy tactic a privacy tactic, because merely changing a certain policy does not by default improve privacy. In PEAR, this attribute is part of the Modifiability tactic group, which we regard a separate quality attribute as described in ISO 25010 [62].
- **Enforce data protection policies, Protect processing**. These tactics are categorized as Enforcement tactics, meaning they are tactics which “include data protection policies enforcement, or processing protection” [40]. Because these tactics are organisational policies, we do not consider them architectural tactics.

Privacy Violation Diagram

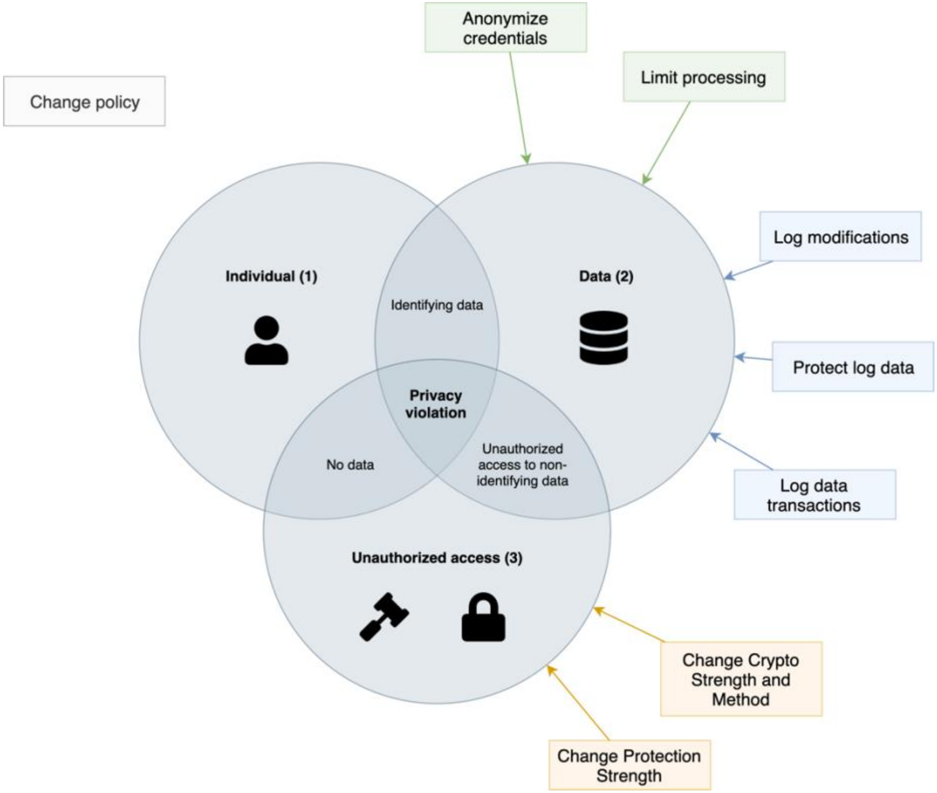


Figure 13: Privacy Enhancing ARchitectures in the Privacy Violation Diagram

Conclusion

The privacy-improving method that is described in PEAR is comparable to the SEI software architecture method as described by Bass et al. and Rozanski and Woods [3, 49], which is the same method we follow to improve privacy. Although the tactic groups proposed by Kung will probably improve privacy in a system to an extent, data awareness and data control tactics are not included.

4.4.4 Tactic selection

In total, 76 privacy tactics were analysed in the last section. The complete collection of tactics can be found in the appendix as Figure 20.

A large amount of these tactics show similarities and overlap. Therefore, we propose a selection of these tactics. As mentioned in the method description in section 4.4.1, we group tactics that show similarities, thereby eliminating doubles. When choosing between specific or more generic tactics, we prefer a larger selection of more specific tactics over fewer generic tactics.

1. Data awareness.

The data awareness tactics we have examined focus on improving data awareness in three different ways: by creating awareness about how data is processed (data processing awareness), what data is processed (content awareness), and breaches (breach awareness). Therefore, corresponding subgroups have been created.

- **Data processing awareness**. CM.AW-P1 Purpose communication (NIST) / Openness, transparency and notice (PRIPARE) / Supply (PS&T).
- **Content awareness**. Notify (PS&T) / CM.AW-P5 Data correction communication (NIST).
- **Breach awareness**. CM.AW-P7 Breach notification (NIST).

Within these subgroups, the proposed tactics are comparable. We therefore propose to pick one tactic per subgroup.

2. Data control.

In this group, we discern two subgroups of tactics: tactics that allow the individual to change its data (change data), and tactics that allow individuals to delete its data (retract data). We choose to name this subgroup 'retract' rather than 'delete', because by retracting data, it gets deleted from the system and implicitly returns to the individual whose property it is.

- **Change data.** CT.DM-P3 Data alteration (NIST) / Update (PS&T)
- **Retract data.** CT.DM-P4 Data deletion (NIST) / Retract (PS&T)

We again propose to pick one tactic per subgroup. We have excluded the Individual participation and access (PRIPARE) tactic and the Intervenability (PG) goals, because they describe the same principles as the two subgroups above, but have grouped it into one more global tactic.

3. Unauthorized access risk mitigation.

For two reasons, we do not recommend to pick from the risk mitigation list of tactics we have analysed. Firstly, we consider tactics in the unauthorized access risk mitigation group to be the most varied by scope of all tactic groups. Very precise tactics to mitigate unauthorized access such as 'Change Crypto strength and Method (PEAR)' therefore are alternated with less precise tactics such as 'PR.DS-P1 Data-at-rest protection (NIST)'. Secondly, as mentioned in section 3.5.1, we see the unauthorized access risk mitigation tactic group as the role of security in privacy. Because the security quality attribute has been researched quite extensively already, we think it more practical to apply a security perspective to structurally address any security concerns in a system. By addressing security, a large part of the privacy 'unauthorized access'-characteristic can be addressed as well.

4. Unauthorized access detection.

By knowing how and when data is accessed and processed, unauthorized access to the data may be discovered. This can be done by logging events and changes to data in the system (logging) and checking whether these events are in line with expected system behaviour (reporting). Data lineage and reconstruction tactics can help with verifying if unauthorized access has occurred (reconstruction). We propose to pick all four tactics for logging, combined with the only reporting and reconstruction tactics.

- **Logging.** CM.AW-P4 Log data disclosures (NIST) / Log data transactions (PEAR) / Protect log data (PEAR) / Log modifications (PEAR)
- **Reporting.** Transparency (PG)
- **Reconstruction.** CM.AW-P6 Maintain data provenance (NIST)

5. Unauthorized access impact mitigation.

In the unauthorized access impact mitigation group, we discern two subgroups: data minimization and unlinkability. Some studies regard unlinkability as part of data minimization. We consider these separate tactics because they both aim to lessen the impact of a privacy violation in different ways.

Our view on data minimization tactics is in line with the three ISO 29100 impact mitigation tactics; data, either identifying or non-identifying, should be collected, retained and disclosed as sparsely as possible and only in accordance with its legitimate purposes [14]. For unlinkability, we use the definition by Pfitzmann and Hansen. "Unlinkability of two or more items of interest means that within the system, the attacker cannot sufficiently distinguish whether these IOIs are related or not." [46].

- **Data minimization**. Exclude (PS&T), Select (PS&T), Strip (PS&T), Destroy (PS&T), Summarize (PS&T), Group (PS&T).
- **Unlinkability**. Mix (PS&T), Obfuscate (PS&T), Isolate (PS&T).

The unauthorized access impact mitigation group is by far the most extensive group and contains 23 tactics. We find that for this group, the Privacy Strategies & Tactics study (PS&T) [12] covers these two subgroups the best. The LINDDUN study focusses only on unlinkability and does not include minimalisation tactics. The NIST framework covers both data minimization and unlinkability reasonably well, but describes the tactics less extensively, making these tactics more difficult to apply in practice. PRIPARE has formulated the tactics extensively, but focusses only on data minimization and not unlinkability. PEAR and Protection Goals are somewhat limited and only contain a combined total of three tactics and are therefore not selected.

6. Consent compliance.

The consent compliance group can be challenging to support as an architect, because most studies include general organisational procedures to ensure consent compliance, rather than tactics that ensure consent compliance by modifying the architecture of the system. This has the interesting result that tactics such as privacy compliance (PRIPARE) are excluded from the selection, because they do not address legal compliance through system architecture.

Consent of the individual is a core property of the ‘unauthorized access’ characteristic. We therefore include tactics that ask for consent for data processing of the individual. Either of the tactics in the group below can be chosen.

- **Consent**. Consent and choice (PRIPARE) / Consent (PS&T) / Policy & consent compliance (LINDDUN).

4.5 Conclusion

In this chapter, we have made a step towards considering privacy as a quality attribute. We have proposed to define the characteristics of the privacy quality attribute as the interdependent collection of the individual, data and unauthorized access. By performing a systematic literature in combination with a grounded theory approach, we have structurally analysed six privacy engineering studies, extracted architectural tactics and mapped these on the three privacy characteristics, thereby creating tactic groups. Because of duplicates and overlap between the tactics, we have made a subselection of tactics that covers all six tactic groups which an architect can use to address privacy in a software system.

5 Evaluating the Quality Attribute

To evaluate the quality attribute for completeness and correctness, we have conducted interviews with experts in the areas of software architecture, privacy and cybersecurity. These participants were chosen because the privacy quality attribute and the privacy tactic groups show similarities and overlap with those research areas. To evaluate whether our approach does not incorrectly interfere with security, we check for correctness. To evaluate whether there are edge cases that do violate privacy but that do not fall in the privacy violation diagram, we check for completeness. The participants were also chosen to check whether our quality attribute approach is compatible with the trade-off process as described in section 2, which falls under our checks for correctness.

The interviews took place via Microsoft Teams and were recorded with permission from the interviewees. The interviews were supported by a presentation, of which the full slide deck can be found in the appendix in section 7.4. The structure of the interview is here first described, followed by the results of the interviews.

After the interviews, some aspects of the privacy quality attribute and the privacy violation timeline were changed. At the time of the interviews, all "unauthorized" tactic groups and the "unauthorized access" characteristic were still called "illegal" and "illegal access" or "data breach". Also, "unauthorized access detection" was still called "data breach discovery".

5.1 Interview structure

Slide 1: Title slide. Study title, authors, date and university.

Slide 2: Describe interview goals. We describe to the interviewee the goal of the interview, which is for them to provide feedback on the completeness and correctness of our main artifact proposals. These are the Privacy Quality Attribute, consisting of the concept of the Privacy Violation Diagram, the resulting Privacy Quality Model, and the tactic groups.

Slide 3: Preliminary information. We meet with the interviewee and determine whether preliminary gathered information is correct. We note their current and past technical roles, years of experience and fields of expertise.

Slide 4: We describe the study goals: to structurally address privacy by applying software architecture principles. We describe the main research question and subquestions.

Slide 5: We describe to the interviewee the steps taken to create the quality attribute and tactics. To do this, we first explain software architecture. We start with the Three Peaks model to position software architecture between requirements engineering and software construction.

Slide 6: We continue explaining the software architecture paradigm, by explaining how the application of perspectives or quality attributes on views can contribute to an improved architecture.

Slide 7: We answer the question whether privacy can be seen as a quality attribute. We conclude that it can, based on current studies. According to ISO 25010 standards however, quality attributes can be described as consisting of characteristics and subcharacteristics.

Slide 8: We find that privacy often is not described as such. This is despite several variants in existing privacy engineering studies. An example is LINDDUNs: 'privacy properties', which we do not consider quality attributes, for reasons on the next slide. Another example is Privacy Strategies and Tactics, which bases itself on GDPR principles rather than privacy characteristics.

We took a strict look at quality attribute composition: a quality attribute should exist of characteristics that together make up the quality attribute. We make a strict distinction between the what and the how and focus on the what. We deem characteristics the what, and tactics to address these characteristics as the how. This means that we do not consider the LINDDUN 'privacy properties' as characteristics, because while 'properties' such as unlinkability may contribute to better privacy, we reckon they represent the how and not the what.

Slide 9: To determine privacy characteristics that adhere to the what principle, we have based ourselves on Westin [9], Solove [52] and ISO 29100 [14] definitions. Based on their work, we have deduced privacy violation characteristics which enables us to split up privacy in three interdependent characteristics: the individual, data, and unauthorized access. We propose these as privacy characteristics. We visualize these characteristics in the Privacy Violation Diagram.

Slide 10: We visualize the Privacy Violation Diagram as a Privacy Quality Model.

Slide 11: Now that we have proposed our privacy quality attribute, we take the next step of addressing quality attribute characteristics by applying architectural tactics; the what. We have therefore gathered tactics from six privacy engineering studies and methods and analysed and grouped these.

Slide 12: As an example, we show wat the LINDDUN study looks like in the Privacy Violation Diagram.

Slide 13: Because of the large amount of tactics in the six studies, we have created six tactic groups, visualized in this slide by applying them on the PVD. We shortly explain how the tactic groups address the three characteristics.

Slide 14: We explain the tactic groups in more detail by showing the privacy violation timeline. Starting on the left, where no data is in the system, we move on to the right describing events that are necessary for a privacy violation to occur. We describe at which point the tactic groups show their relevancy.

Slide 15: We show the Privacy Quality Model, consisting of characteristics with the addition of the tactic groups.

Slide 16: We ask the interviewee to evaluate the correctness and completeness of the Privacy Violation Diagram, Privacy Quality Model, and Tactic Groups.

Slide 17: Bibliography 1-2

Slide 18: Bibliography 2-2

As mentioned, we are mainly interested in the completeness and correctness of our approach. We will however include other relevant feedback if provided by the interviewee. No interview information is gathered quantitatively due to the difficulty of measuring of the concepts of completeness and correctness.

5.2 Participants

The following interviewees have participated in the evaluation interviews. We list their current roles, areas of expertise and the date and time of the interviews.

Figure 14: interviewees' roles, expertise and interview time and date

Participant	**Role**	**Expertises**	**Interviewed**
P1	University professor, subjects: Privacy, Security and IT law	Privacy by Design, Privacy & IOT, Privacy & Identity Management	April 28, 2021, 13:00 – 15:00
P2	Cyber Security Specialist	Cyber Security	April 30, 2021, 09:00 – 09:45
P3	CTO, Software Architecture expert	Software Engineering, Software Architecture	May 4, 2021, 10:00 – 10:35
P4	Cyber Security Analyst	Cyber Security	May 10, 2021, 20:30 – 21:30

5.2.1 Participant 1

Privacy Violation Diagram and Quality Model

Completeness

P1 compares the PVD to the GDPR and notes some edge cases where the PVD does not seem to be entirely complete. He gives the examples of the right to be informed and the right of access, as determined in the GDPR. We indicate these can be covered by tactics that address the characteristics, if the architect wishes to do so. We note there is a key difference in the PVD approach versus that of Colesky and Hoepman [11]; whereas Colesky and Hoepman base their privacy design strategies and tactics on the GDPR, the PVD bases itself on definitions by Solove, Westin and ISO 27000. In P1's opinion, this limits the applicability of the PVD to information privacy. We reply that the thesis contains a section that indeed limits the thesis to information privacy.

Correctness

P1 initially doubts the usefulness of adding the additional layer of characteristics between privacy and tactics, because in his experience, this mostly adds to the general confusion that surrounds privacy. He also doubts whether the characteristics are as interdependent as we initially have presented it. To answer these doubts, we go back to the PVD and give the example of the Individual who retracts himself from the situation, after for example having been influenced by data awareness tactics. By the retraction of the Individual, we pose that a privacy violation is averted, thus supporting the concept of interdependency. P1 replies that, while this example may work, he feels that edge cases exist where the interdependency notion does not hold. As an example, he changes the situation and gives the example of an Individual not being in the system by its own free will, and wonders how this changes the situation. We pose that because the Individual is not in the system by its own free will, its identifying data is processed illegally, thus satisfying the three characteristics of a privacy violation.

We move on to another example, where P1 gives the example of a customer being in the database of a large online retailer, while the retailer has not implemented the option for the customer to change its data in the database of the retailer. P1 states that, according to the PVD, this is not a privacy violation, while the GDPR categorizes it as one. We repeat the statement and ask whether P1 considers not having the option to change your own data a privacy violation, which he confirms. In the interview, we respond by restating that we do not aim to be GDPR-compliant in our approach, because we aim for our method to be usable outside of the EU as well.

Lastly, P1 asks how the PVD relates to IoT-devices such as a smart thermostat. We discuss the scenario when the thermostat is manipulated by a third party, causing temperature changes in the house of the thermostat owner. P1 considers this an invasion of privacy which does not really require data, which therefore is not covered by the PVD.

Privacy Tactic Groups

Completeness

P1 does not spot any immediately obvious missing tactic groups and therefore considers the tactic groups to be reasonably complete.

Correctness

P1 notices we have created the groups data breach risk mitigation, data breach discovery and data breach impact mitigations, and suggests it would be better to rename the ‘data breach’ component to ‘illegal access’, because data breach is more specific than illegal access. We agree on this. He asks whether illegal access does not by default mean that a privacy violation has occurred, to which we reply that it depends on whether identifying data is accessed, to which he agrees.

5.2.2 Participant 2

Privacy Violation Diagram and Quality Model

Completeness

P2 does not have further comments on the completeness of the model and focusses on correctness.

Correctness

With regards to the illegal access characteristic, P2 thinks it might be possible to violate an individual's privacy in a legal way. To this, we reply that the illegal access characteristic consists of two parts; illegal either by law or because of lack of consent of the individual. P2 thinks it to be right to integrate law in the PVD this way.

P2 also asks whether it is right that the PVD means that, when a data processing organisation chooses to ignore a data leak, this would mean the privacy violation has not occurred. As an example, he names the allekabels.nl data leak from February 2021, which allekabels.nl attempted to hush up. We reply that the PVD should be read differently, because by hushing up the data leak, the violation itself has still occurred. We also say that our main angle of approach is the good faith of the architect and the data processor and suggest illegal access detection tactics to improve this specific issue.

Privacy Tactic Groups

Completeness

The tactic groups are considered complete and logical by P2. He also is interested in the illegal access discovery tactic group, thinking that much work can still be done in this field to do this in a privacy-friendly way. He ways it is difficult not to violate the privacy of the individual when deploying illegal access discovery methods. We reply by asking whether grouping and summarizing tactics might be helpful in this scenario, because these tactics shield off the individual in the collection. He agrees these may be helpful.

Correctness

P2 wonders whether the order of the illegal access impact mitigation group on the privacy violation timeline is correct; he would have expected to find it earlier on in the process. We explain that, while the tactics in this group (such as data minimisation) should indeed be considered as one of the first when building the system because they are difficult to change later on, they mainly show their relevance later on in the process towards a privacy violation. After explaining, P2 considers this correct.

5.2.3 Participant 3

Privacy Violation Diagram and Quality Model

Completeness

P3 states that if the PVD is not fully GDPR compliant, that might make the PVD somewhat less useful, because most engineers are mainly interested in being GDPR compliant. He thus suggests to describe which parts of the GDPR are not covered by the PVD. P3 says that would make the PVD more useful, because then an architect can consider and make decisions based on that.

Correctness

P3 says he originally thought of privacy as a subset of security. He reasons that if privacy is not a subset, it needs to contain some elements that security does not. P3 sees security as whether data can only be accessed by those who are authorized to accessed it. He continues his reasoning and says that whether it concerns identifying data or not is irrelevant to security. He concludes that privacy does seem to have some interesting characteristics which are not part of security. P3 says he is convinced that the relation between security and privacy as visualized in the PVD is right and sees it is an elegant approach, despite the fact that there is the slight awkwardness when formulating privacy as the absence of a privacy violation.

Privacy Tactic Groups

Completeness

P3 cannot spot any immediate missing tactics, partially because he thinks the tactic groups are quite abstract. It seems to him that the tactic groups are more complete than '90% of what that software architects are used to'.

Correctness

P3 thinks it right to use the term 'illegal access' rather than data breach, because data breach is quite broad and somewhat inaccurate. Besides this, he does not have any immediate objections to the method we propose. He does note he is not an expert on the subject of privacy.

5.2.4 Participant 4

Privacy Violation Diagram and Quality Model

Completeness

P4 asks if we have discern any gradation in severity of privacy violations. We answer that we discern the steps towards a privacy violation in the privacy violation timeline; the first step being that data is secured in the system, the second step being illegal access, and the third step being a privacy violation. Besides those steps, we have purposefully not created a finer gradation, because while it make the model more complete, it also leads to higher complexity which can limit the usability of the PVD.

Correctness

P4 focusses on the relation between security and privacy, and wonders what the differences are between those two. He states that most privacy characteristics have to do with security, such as discovery and impact mitigation. He asks whether privacy in essence is not a security issue. We reply that security indeed plays a large role, because by keeping data secure, no privacy violation can take place in the first place. However, we regard tactics such as data minimalisation and thereby impact mitigation as distinct privacy tactics. Like participant 3 mentioned, security is mainly about determining that the right person gets access to the right resources. It rarely occupies itself with the content of those resources, which privacy does.

P4 also asks if we discern a gradation of identifying data, because a first and last name can be used more easily for identification than a cookie or IP address. We reply that we do recognize these differences, but that they are not included in the PVD.

Privacy Tactic Groups

Completeness and Correctness

P4 considers the privacy violation timeline and the PVD to be good models that are useful to support a discussion about privacy. He thinks the best aspect of the model is the clearness and conciseness of the three characteristics, which allows novices and experts to reason about the characteristic.

P4 asks whether we have considered a scoring system, where we would examine a system or study and assess it based on the amount of tactics applied or the amount of characteristics that are addressed. We reply that have not yet, but might suggest it in the future work section.

5.3 Results

In general, the participants are positive about the correctness and completeness of the privacy violation diagram, privacy violation timeline and privacy tactic groups. P1 formulates some edge cases with regards to the correctness of the PVD and also asks questions about GDPR compliancy. Whether these are critical to the correctness of our approach is elaborated upon in the discussion.

P2 wonders whether the order of the tactic groups in the privacy violation timeline is correct, because in his view, impact mitigation tactics do not by definition have to come last in the sequence. We again elaborate upon this in the discussion. The proposed relationship between security and privacy is somewhat new to interviewees P1 and P3, although they consider it valid. P3 also confirms that, in his view, the privacy quality model is correct and suitable to use in the software architecture trade-off process.

The privacy tactic groups are generally considered complete and correct, with the side note that due to its abstractness, application of those groups may present some challenges. It is also indicated by participants that a strength of the quality attribute is that it allows stakeholders to reason about the quality attribute.

5.4 Discussion

P1 states that if an individual does not have the option to change its data, this is considered a privacy violation by the GDPR. Despite this, we think the basic assumptions of the PVD still hold: if an individual does not have the option to change its data, this is not necessarily a privacy violation. This does not mean the event should not be addressed. It also illustrates how a GDPR clause such as this fits in the PVD approach: by giving the individual the option to change its data, the GDPR effectively prescribes an architectural and procedural tactic to help the individual with safeguarding its privacy. This again emphasizes the need to distinguish between a tactic and characteristic for an architect.

Another example is the thermostat example given by P1. P1 considers the thermostat hack a privacy violation that is not covered by the PVD. We think this scenario is partially covered by the PVD, because there is a specific individual (1) whose thermostat is accessed without authorization (3) for which it is necessary to possess identifying information (2) to identify and target the individual who owns the thermostat. What makes this particular scenario an edge case, is the fact that the identifying information is not accessed without authorization, but has to be obtained in advance to access the thermostat. We consider this example an edge case, closer to physical privacy and security than to privacy. In our view, this example demonstrates that security and privacy are knit together tightly and that it can be difficult to distinguish one from the other.

Our last discussion point is the connection between the GPDR and the Privacy Violation Diagram. The Privacy Violation Diagram may not immediately lead to GDPR compliancy, because that is not its goal. Comparably, implementing the security quality attribute does not directly lead to ISO 27001 compliancy. The method of applying quality attributes and perspectives allows the architect to go as far as he likes in satisfying its stakeholders requirements. The tactic groups may support GDPR compliance, if the architect picks the right tactics to reach GDPR compliance. This decision is up to the architect and the stakeholders of the project.

6 Conclusion, limitations and future work

6.1 Conclusion

This thesis attempts to embed privacy in software architecture by formulating it as a quality attribute. It is structured around the following research question: 'How to effectively embed privacy as a quality attribute in software architecture?'. The answer to this is formulated by answering the following three subquestions.

The answer to subquestion 1, 'How can privacy be regarded as a quality attribute', is formulated by performing literature research about the structure of a quality attribute, privacy developments and the software architecture trade-off process. It shows that privacy can be embedded in the software architecture process as quality attribute by defining its characteristics, thereby forming the basis for an architectural perspective on privacy. An architectural perspective would allow privacy to be applied orthogonally on architectural views, thereby providing the software architect with a way of addressing privacy concerns in an architecture design. This quality attribute is a first step towards this perspective.

Based on information privacy definitions, we propose a privacy quality attribute consisting of three characteristics: the individual, data, and unauthorized access. We think these characteristics to be interdependent for a privacy violation to occur, meaning that by addressing just one characteristic, a privacy violation can be prevented. However, to lower the risk of a privacy violation, we recommend the architect to address all characteristics. The three privacy characteristics are summarized in the proposed privacy violation diagram in Figure 4.

By addressing privacy in the architecture design phase of software construction, our approach supports the core concept in Privacy by Design: that privacy should be addressed in software design from the outset. Principles that are supported by default are 'Proactive not reactive – Preventative, not Remedial' and 'Privacy Embedded into Design'. For other principles, such as 'Privacy as the default', 'Full functionality' and 'End-to-end-security', it is up to the architect and the tactics he chooses to which extent he supports those. For example, if the architect chooses to sacrifice some privacy aspects for functionality, he does not necessarily adhere to the fourth foundational principle. These decisions are left up to the architect.

Subquestion 2, ‘How can current privacy conceptualizations be harmonized into a quality attribute?’, is answered by performing a systematic literature view combined with a grounded theory approach. We categorise architectural tactics from existing privacy engineering studies to create tactics group that address the three characteristics. We propose six tactic groups, called data awareness, data control, unauthorized access risk mitigation, unauthorized access detection, unauthorized access impact mitigation, and consent compliance.

We note that many existing privacy engineering methods contain mixes of architectural characteristics and architectural tactics. We therefore attempt to clearly divide our proposed privacy characteristics and tactic groups. This separation is visualised in the privacy violation timeline in Figure 7, demonstrating how our proposed tactic groups may prevent a privacy violation during each step towards a violation.

Besides the proposal for tactic groups, a final selection of tactics was made. We propose this selection as a basis to work towards a complete architectural perspective on privacy.

Subquestion 3, ‘What is the perceived completeness and correctness of the quality attribute?’ has been answered by performing interviews with experts in the areas of privacy, security and software architecture. In the privacy violation diagram, security aspects are incorporated as part of the unauthorized access risk mitigation tactic group, which experts agree on is a valid approach. It shows that security and privacy are inherently linked, because security tactics are essential to protect privacy, but that security tactics alone may not be enough to prevent a privacy violation. Experts also note that the privacy violation diagram help with facilitating a structured discussion about privacy.

With regards to the GDPR, experts note that the privacy violation diagram does not directly lead to GDPR compliance. To facilitate GDPR compliance in software architecture, we propose a section in the architectural perspective, to ensure that all GDPR clauses are implemented by the appropriate architectural tactics.

6.2 Limitations

This study proposes only a first step towards embedding privacy in software architecture, by formulating privacy as a quality attribute. The practical applicability is therefore somewhat limited, because the quality attribute has not yet been expanded to a complete architectural perspective. To improve practical applicability, an architectural perspective based on this quality attribute can be created in the future.

Another limitation is that this study solely focusses on software architecture and therefore excludes organisational procedures. This means that implementing the privacy quality attribute as we propose it does not directly lead to GDPR compliancy for the stakeholders of the software system. As discussed in the evaluation section, this is also not the goal of the attribute. We are positive that implementing the right tactics of the privacy quality attribute can lead to a GDPR compliant software system. However, for an organisation to reach GDPR compliancy as a whole, additional organisational policies are probably necessary.

Regarding the architectural tactic selection, even though the analysis of tactics is quite extensive, the analysis could have been more in-depth. The large amount of tactics in the privacy engineering studies has not permitted it to comment and analyse every tactic, such as with the NIST Privacy Framework, where we have only been able to support our reasons for inclusion of the tactics and not exclusion.

Another limitation is the fact that there is little to no consensus about what an 'architectural tactic' precisely is. We have discussed this problem in section 2.5 and also commented on this issue for every method we have analysed. The issue remains however that relatively precise tactics such as mix ("create mix networks that reduce correlation between different pieces of personal data" [12]) are analysed and compared to relatively abstract tactics, such as integrity ("the need for reliability and non-repudiation regarding a given piece of information, i.e. the need for processing unmodified, authentic, and correct data") [32]. This mainly causes challenges when selecting tactics without causing overlap. We therefore have given an elaborate explanation for why we choose each tactic.

This study is also limited by the fact that, even though we have been as extensive as possible in our analysis, privacy is a culturally dependent concept that often changes through time. We have been exhaustive to the best of our efforts, but cannot guarantee full coverage of every aspect of information privacy.

6.3 Future work

In this thesis, a quality attribute for privacy is created. The quality attribute forms the necessary groundwork for developing a more extensive way of addressing privacy, which is to formulate an architectural perspective. As mentioned in section 2.4, a perspective not only consists of architectural tactics and a quality attribute; it often includes sections on view applicability, describes concerns, contains activities or method steps, describes problems and pitfalls and sometimes suggests checklists as well. Creating a more complete perspective on privacy would therefore be a great way of creating a more structured approach to address privacy. Especially the LINDDUN, PS&T and PEAR studies provide method steps that reasonably easy to follow, which help greatly with formulating the method steps and activities in a privacy perspective. In our view, the construction of the architectural perspective would be a prime candidate for further research.

Another way of expanding on this thesis would be to precisely formulate the differences between GDPR provisions and the quality model as we present it. From the start, we have decided not to directly translate GDPR requirements to software, but to choose the SEI software architecture approach. Even though we think our approach to be reasonably complete, we cannot guarantee GDPR compliancy because this is up to the architect. An analysis of which architectural tactics need to be applied to ensure GDPR compliance would therefore be useful. It would allow the architect to better assess which areas he needs to double check, when using our artefacts.

With regards to the expert interviews, we only do a partial evaluation that focusses on correctness and completeness. However, what we also need to know is whether a privacy perspective based on the proposed privacy quality attribute truly improves how architects address privacy. This has not been performed due to time constraints, but could be a valuable addition in further research.

Lastly, we feel that the privacy tactic groups as we propose them still are quite abstract and could benefit from further development. When creating the tactic selection in section 4.4.4, we already noted that several subselections can be made quite easily. A further investigation in subselections of privacy tactic groups may therefore be beneficial to create a more complete privacy quality attribute and privacy perspective.

Bibliography

1. Bachmann, F. et al.: Tactics : A Step Toward Tactics : A Step Toward. (2003).
2. Barbacci, M.R. et al.: Analysis Method (CBAM). (2003).
3. Bass, L. et al.: Software Architecture in Practice. (2012).
4. Brinkkemper, S., Pachidi, S.: Functional architecture modeling for the software product industry. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 6285 LNCS, August, 198–213 (2010). https://doi.org/10.1007/978-3-642-15114-9_16.
5. Burgoon, J.K.: Privacy and Communication. Ann. Int. Commun. Assoc. 6, 1, 206–249 (1982). https://doi.org/10.1080/23808985.1982.11678499.
6. Cambridge Dictionary: INDIVIDUAL - meaning in the Cambridge English Dictionary.
7. Cavoukian, A.: Privacy by Design - The 7 foundational principles - Implementation and mapping of fair information practices. Inf. Priv. Comm. Ontario, Canada. 5 (2009). https://doi.org/10.1007/s12394-010-0062-y.
8. Charmaz, K.: The Search for Meaning - Grounded Theory. Grounded Theory. Rethinking Methods in Psychology, 27–49 (1996).
9. Clark, T.C., Westin, A.F.: Privacy and Freedom. Calif. Law Rev. 56, 3, 911 (1968). https://doi.org/10.2307/3479272.
10. Clements, P. et al.: Documenting Software Architectures. Addison-Wesley (2010).
11. Colesky, M. et al.: A Critical Analysis of Privacy Design Strategies. Proc. - 2016 IEEE Symp. Secur. Priv. Work. SPW 2016. May 2016, 33–40 (2016). https://doi.org/10.1109/SPW.2016.23.
12. Colesky, M. et al.: A Critical Analysis of Privacy Design Strategies. Proc. - 2016 IEEE Symp. Secur. Priv. Work. SPW 2016. 33–40 (2016). https://doi.org/10.1109/SPW.2016.23.
13. Commission, E.: What is a data controller or a data processor?, https://ec.europa.eu/info/law/law-topic/data-protection/reform/rules-business-and-organisations/obligations/controller-processor/what-data-controller-or-data-processor_en, last accessed 2018/12/04.

14. Commission, I.E., Standardization, I.O. for: ISO 29100: Information technology — Security techniques — Privacy framework Technologies. 2011, 1–22 (2011).
15. Dienlin, T.: The privacy process model. Medien und Priv. [Media privacy]. January 2014, 105–122 (2014).
16. DutchNews: None of seven proposed corona apps meets privacy criteria, says legal advisor, https://www.dutchnews.nl/news/2020/04/none-of-seven-proposed-corona-apps-meets-privacy-criteria-says-legal-advisor/, (2020).
17. EU: First GDPR proposal, https://eur-lex.europa.eu/legal-content/en/ALL/?uri=CELEX%3A52012PC0010.
18. EU: What are the GDPR Fines?, https://gdpr.eu/fines/, last accessed 2018/11/20.
19. Europe, C. of: Chart of signatures and ratifications of Treaty 108, https://www.coe.int/en/web/conventions/full-list/-/conventions/treaty/108/signatures?p_auth=ed9HoM5Q, last accessed 2018/12/02.
20. European Commission: Better Regulation Guidelines SWD. (2017).
21. European Commission: Does my company/organisation need to have a Data Protection Officer (DPO)? Eur. Comm. (2017).
22. European Data Protection Supervisor: Preliminary Opinion on privacy by design. May, 34 (2018).
23. European Parliament and of the Council: Regulation (EU) 2016/679 of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). Off. J. Eur. Communities. OJ L 119/1, 1–88 (2016).
24. Fox, M.: Alan F. Westin, Who Transformed Privacy Debate Before the Web Era, Dies at 83. New York Times. (2013).
25. Fretten, C., Miller, V.: The European Union : a guide to terminology , procedures and sources. July, 1–16 (2005).
26. Gellman, R.: I . Origins of FIPs. 1–24 (2012).
27. Goddard, M.: Viewpoint: The EU general data protection regulation (GDPR): European regulation that has a global impact. Int. J. Mark. Res. 59, 6, 703–706 (2017). https://doi.org/10.2501/IJMR-2017-050.
28. Guardian, T.: Norway suspends virus tracing app due to privacy concerns, https://www.theguardian.com/world/2020/jun/15/norway-suspends-virus-tracing-app-due-to-privacy-concerns, (2020).

29. Gürses, S., Del Alamo, J.M.: Privacy Engineering: Shaping an Emerging Field of Research and Practice. IEEE Secur. Priv. 14, 2, 40–46 (2016). https://doi.org/10.1109/MSP.2016.37.

30. Gürses Seda, Troncoso Carmela, D.C.: Engineering: Privacy by design. IMDEA Softw. (2011).

31. Hafiz, M.: A pattern language for developing privacy enhancing technologies. Softw. Pract. Exp. 43, 7, 769–787 (2013). https://doi.org/10.1002/spe.1131.

32. Hansen, M. et al.: Protection goals for privacy engineering. Proc. - 2015 IEEE Secur. Priv. Work. SPW 2015. 159–166 (2015). https://doi.org/10.1109/SPW.2015.13.

33. Hartzog, W.: THE INADEQUATE , INVALUABLE FAIR INFORMATION PRACTICES. Maryl. Law Rev. 76:952, (2017).

34. Hiller, J.S., Russell, R.S.: Privacy in Crises: The NIST Privacy Framework. J. Contingencies Cris. Manag. 25, 1, 31–38 (2017). https://doi.org/10.1111/1468-5973.12143.

35. Jansen, A., Bosch, J.: Software architecture as a set of architectural design decisions. Proc. - 5th Work. IEEE/IFIP Conf. Softw. Archit. WICSA 2005. 2005, May, 109–120 (2005). https://doi.org/10.1109/WICSA.2005.61.

36. Kazman, R. et al.: ATAM: SM Method for Architecture Evaluation Product Line Systems. August, (2000).

37. Kitchenham, B., Charters, S.: Guidelines for performing Systematic Literature Reviews in Software Engineering}, (2007). https://doi.org/10.1.1.117.471.

38. Kruchten, P.: Architectural Blueprints---The ``4+ 1" View Model of Software Architecture. Tutor. Proceedings, Tri-Ada'95. 12, November, 540–555 (1995).

39. Kruchten, P. et al.: The past, present, and future for software architecture. IEEE Softw. 23, 2, 22 (2006). https://doi.org/10.1109/MS.2006.59.

40. Kung, A.: PEARs: Privacy Enhancing ARchitectures. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 8450 LNCS, 18–29 (2014). https://doi.org/10.1007/978-3-319-06749-0_2.

41. Le, D. et al.: PRIPARE Methodology Handbook. (2015).

42. NIST: NIST Privacy Framework: a Tool for Improving Privacy Through Enterprise Risk Management. (2020).

43. Notario, N. et al.: PRIPARE: Integrating privacy best practices into a privacy engineering methodology. Proc. - 2015 IEEE Secur. Priv. Work. SPW 2015. 151–158 (2015). https://doi.org/10.1109/SPW.2015.22.

44. OECD: The OECD Glossary of Statistical Terms 2008. (2008).

45. Parnas, D.L.: On the Criteria to Be Used in Decomposing Systems into Modules. Pioneers Their Contrib. to Softw. Eng. 479–498 (1972). https://doi.org/10.1007/978-3-642-48354-7_20.

46. Pfitzmann, A., Hansen, M.: A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management. Tech. Univ. Dresden. 1–98 (2010). https://doi.org/10.1.1.154.635.

47. Ramirez, E.: Privacy By Design and the New Privacy Framework of the U.S. Federal Trade Commission, (2012).

48. Rozanski, N., Woods, E.: Software systems architecture: working with stakeholders using viewpoints and perspectives. Addison-Wesley (2012).

49. Rozanski, N., Woods, E.: Software Systems Architecture. In: Software Systems Architecture. (2012).

50. Senarath, A., Arachchilage, N.A.G.: Understanding Software Developers' Approach towards Implementing Data Minimization. 13–16 (2018).

51. Solove, D.J.: A Taxonomy of Privacy. Univ. PA. Law Rev. 154, 3, 477 (2006). https://doi.org/10.2307/40041279.

52. Solove, D.J.: Conceptualizing Privacy. Calif. Law Rev. 90, 4, 1087 (2002). https://doi.org/10.2307/3481326.

53. Spiekermann, S.: The challenges of privacy by design. Commun. ACM. 55, 7, 38–40 (2012). https://doi.org/10.1145/2209249.2209263.

54. Spiekermann, S., Cranor, L.F.: Engineering Privacy. IEEE Trans. Softw. Eng. 35, 1, 67–82 (2009). https://doi.org/10.1109/TSE.2008.88.

55. U.S. Dep't. of Health, Education and Welfare, Secretary's Advisory Committee on Automated Personal Data Systems, Records, computers, and the R. of C.: 1973: The Code of Fair Information Practices. (1973).

56. Voigt, P., von dem Bussche, A.: The EU General Data Protection Regulation (GDPR). Springer International Publishing, Cham (2017). https://doi.org/10.1007/978-3-319-57959-7.

57. Warren, S., Brandeis, L.: The Right to Privacy. Harv. Law Rev. IV, 5, (1890).

58. van de Weerd, I., Brinkkemper, S.: Meta-modeling for situational analysis and design methods. Handb. Res. Mod. Syst. Anal. Des. Technol. Appl. 35–54 (2008).

https://doi.org/10.4018/978-1-59904-887-1.ch003.

59. Wieringa, R.: Design science methodology. (2010). https://doi.org/10.1145/1810295.1810446.

60. Woods, E., Rozanski, N.: Using architectural perspectives. Proc. - 5th Work. IEEE/IFIP Conf. Softw. Archit. WICSA 2005. 2005, 25–34 (2005). https://doi.org/10.1109/WICSA.2005.74.

61. Wuyts, K.: LINDDUN : a privacy threat analysis framework. (2014).

62. ISO/IEC/IEEE 25010: Systems and software quality models. (2011).

63. The California Consumer Privacy Act of 2018. California State Legislature (2018).

7 Appendix

7.1 Colour-coded overview of tactics per study

We provide the following overview to further structure the tactics. The colours match the tactic groups defined in section 4.4.2. We do not consider the tactics with the colour black as architectural privacy tactics, for reasons elaborated upon in the next section.

Figure 15: Colour-coded overview of privacy tactics.

LINDDUN	Unlinkability	Anonymity	Pseudonymity	Plausible deniability	Undetectability	Confidentiality	Content awareness	Policy & consent compliance
NIST Privacy Framework	CT.DM-P1 Data review	CT.DM-P2 Data disclosure	CT.DM-P3 Data alteration	CT.DM-P4 Data deletion	CT.DP-P1 Limit observability and linkability	CT.DP-P2 Limit identification of individual	CT.DP-P3 Limit inferences	CT.DP-P4 Selective collection
	CT.DP-P5 Attribute substitution	CM.AW-P1 Purpose communication	CM.AW-P3 Visibility	CM.AW-P4 Log data disclosures	CM.AW-P5 Data correction communication	CM.AW-P6 Maintain data provenance	CM.AW-P7 Breach notification	CM.AW-P8 Individual mitigation
	PR.AC-P1 Credentials	PR.AC-P4 Authorization	PR.AC-P5 Network segregation	PR.DS-P1 Data-at-rest protection	PR.DS-P2 Data-in-transit protection			
Privacy Strategies & Tactics	Exclude	Select	Strip	Destroy	Restrict	Mix	Obfuscate	Dissociate
	Distribute	Isolate	Summarize	Group	Supply	Notify	Explain	Consent

	Choose	Update	Retract	Create	Maintain	Uphold	Audit	Log
	Report							
PRIPARE	Consent and choice	Purpose legitimacy and specification	Collection limitation	Data minimization	Use, retention and disclosure limitation	Accuracy and quality	Openness, transparency and notice	Individual participation and access
	Accountability	Information Security	Privacy compliance					
Protection Goals	Confidentiality	Integrity	Availability	Transparency	Unlinkability	Intervenability		
PEAR	Anonymize credentials	Limit processing	Enforce data protection policies	Protect processing	Log data transactions	Log modifications	Protect log data	Change policy
	Change Crypto Strength and Method	Change Protection Strength						

7.2 Process-deliverable diagrams

7.2.1 LINDDUN

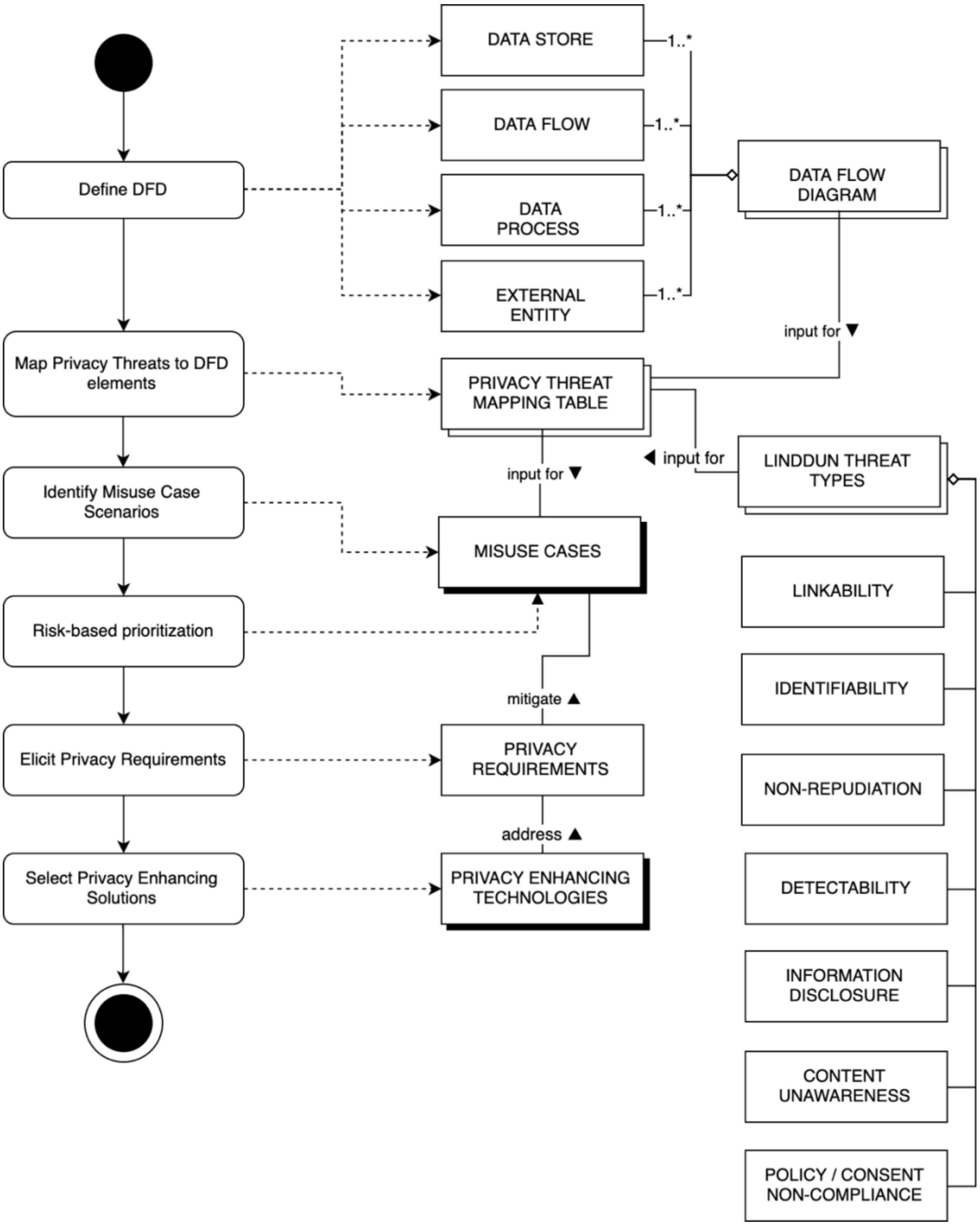
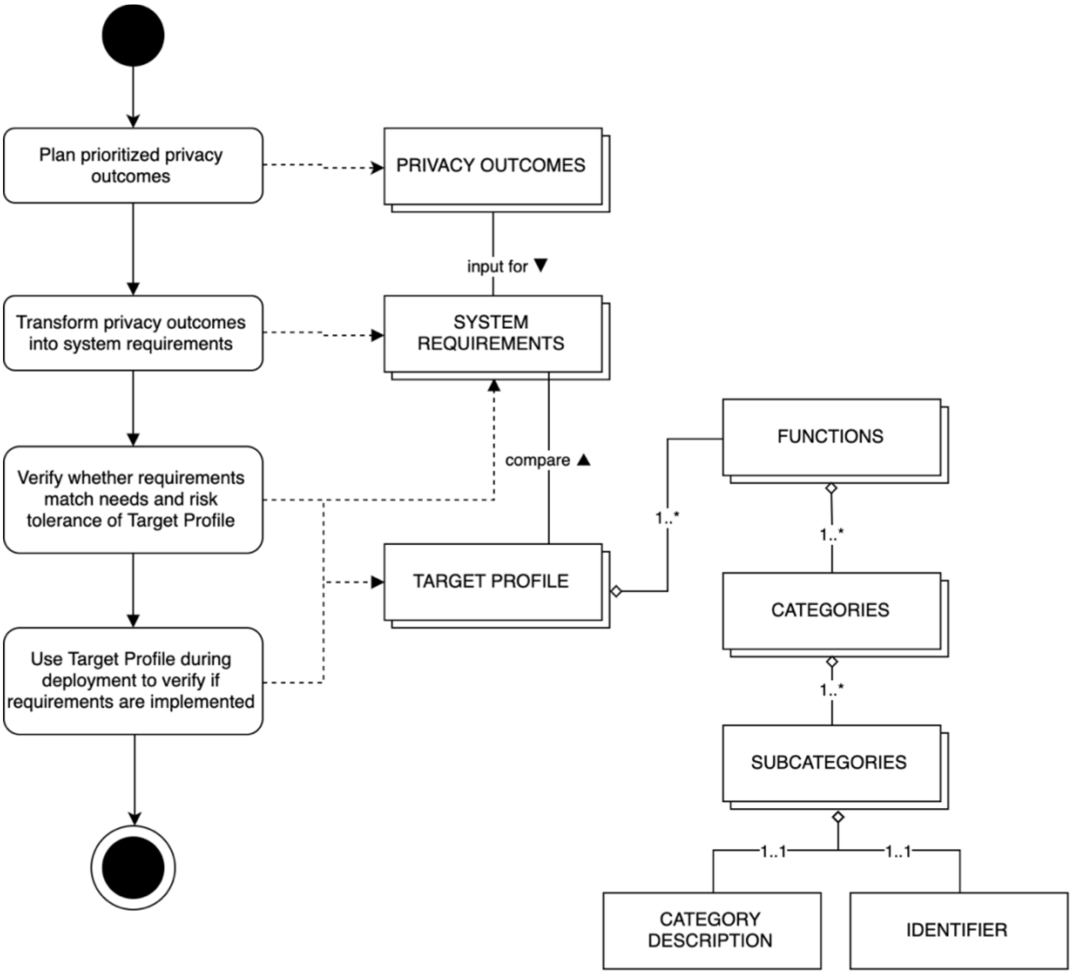


Figure 16: Process-Deliverable Diagram of the LINDDUN method

7.2.2 NIST



Plan prioritized privacy outcomes
PRIVACY OUTCOMES
input for ▼
Transform privacy outcomes into system requirements
SYSTEM REQUIREMENTS
compare ▲
Verify whether requirements match needs and risk tolerance of Target Profile
TARGET PROFILE
FUNCTIONS
1..*
1..*
CATEGORIES
1..*
Use Target Profile during deployment to verify if requirements are implemented
SUBCATEGORIES
1..1
1..1
CATEGORY DESCRIPTION
IDENTIFIER

Figure 17: Process-Deliverable Diagram of the NIST Privacy Framework

7.2.3 Privacy Strategies and Tactics

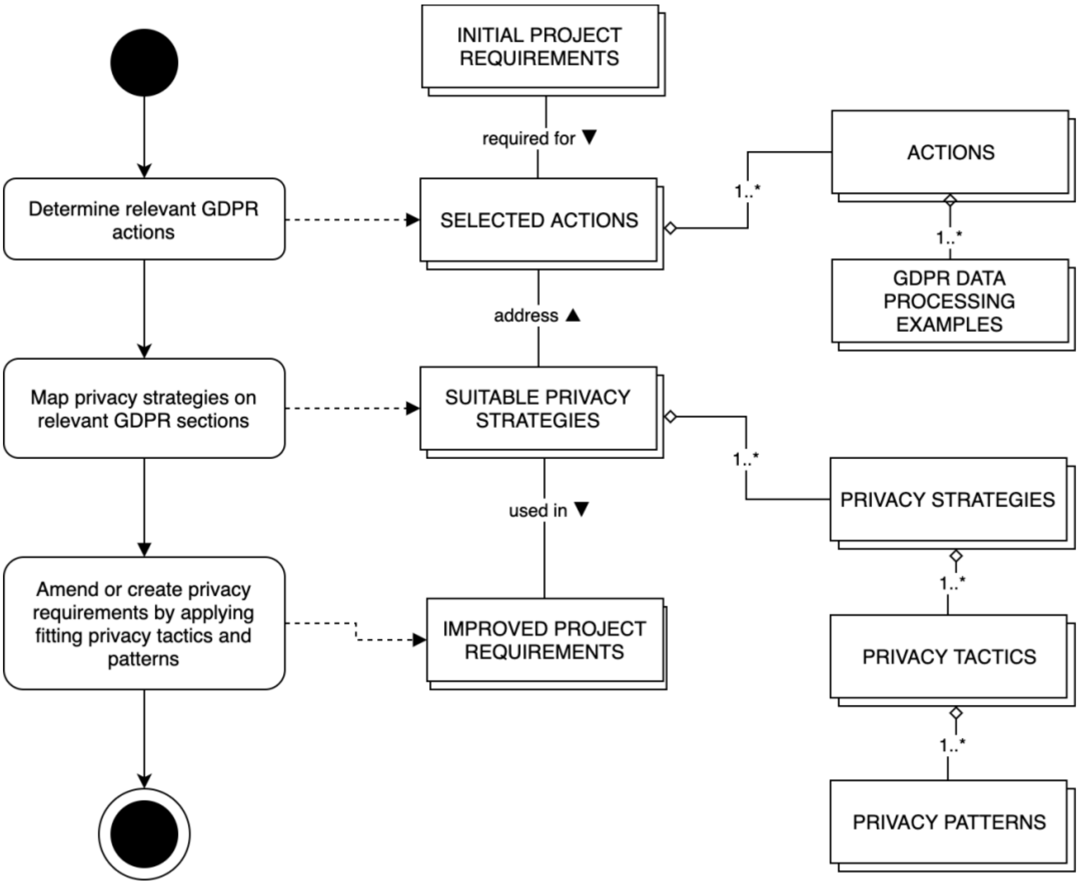

INITIAL PROJECT REQUIREMENTS
required for ▼
Determine relevant GDPR actions
SELECTED ACTIONS
1..*
ACTIONS
1..*
GDPR DATA PROCESSING EXAMPLES
address ▲
Map privacy strategies on relevant GDPR sections
SUITABLE PRIVACY STRATEGIES
1..*
PRIVACY STRATEGIES
used in ▼
1..*
Amend or create privacy requirements by applying fitting privacy tactics and patterns
IMPROVED PROJECT REQUIREMENTS
PRIVACY TACTICS
1..*
PRIVACY PATTERNS

Figure 18: Process-Deliverable Diagram of the PS&T study

7.2.4 PRIPARE

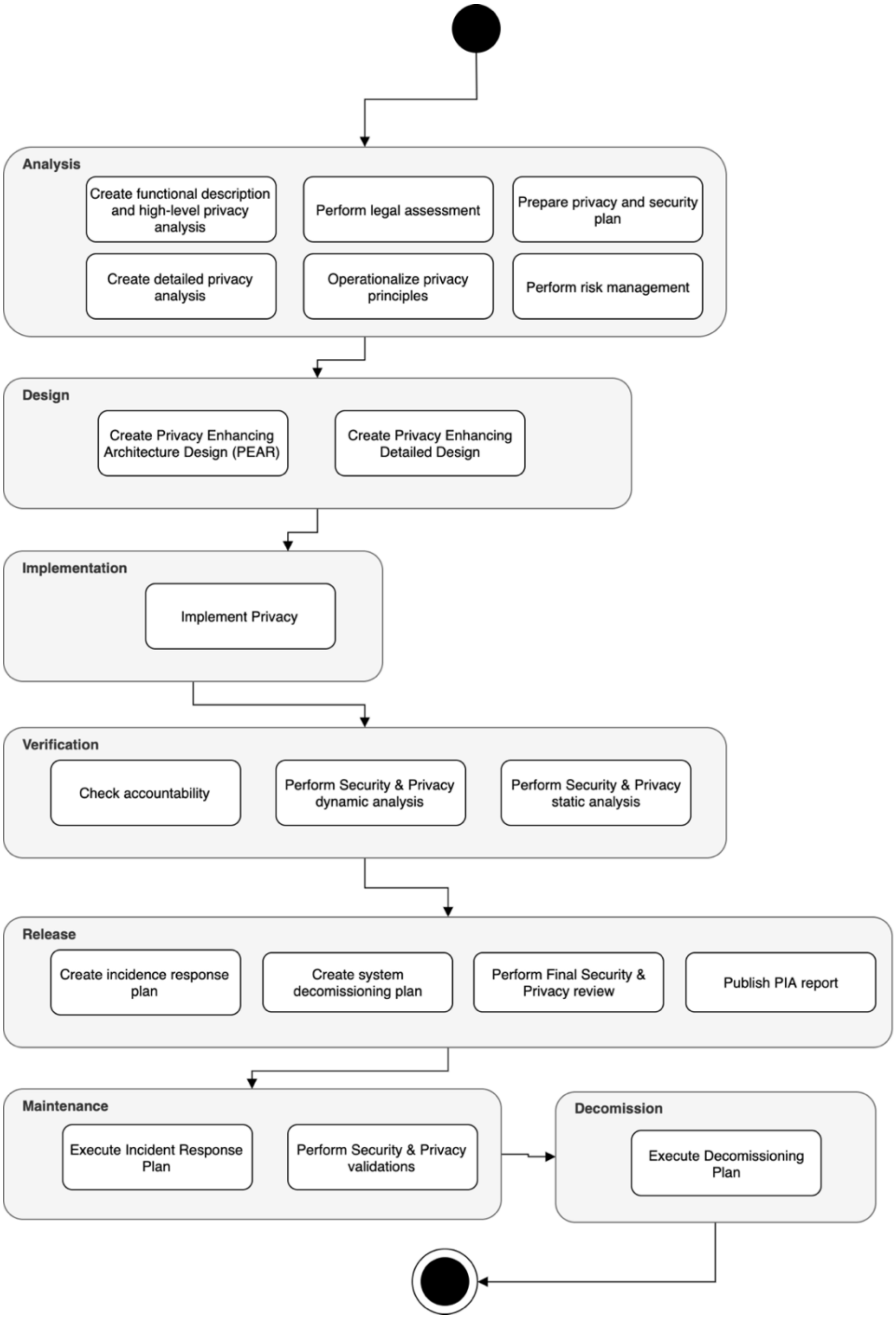
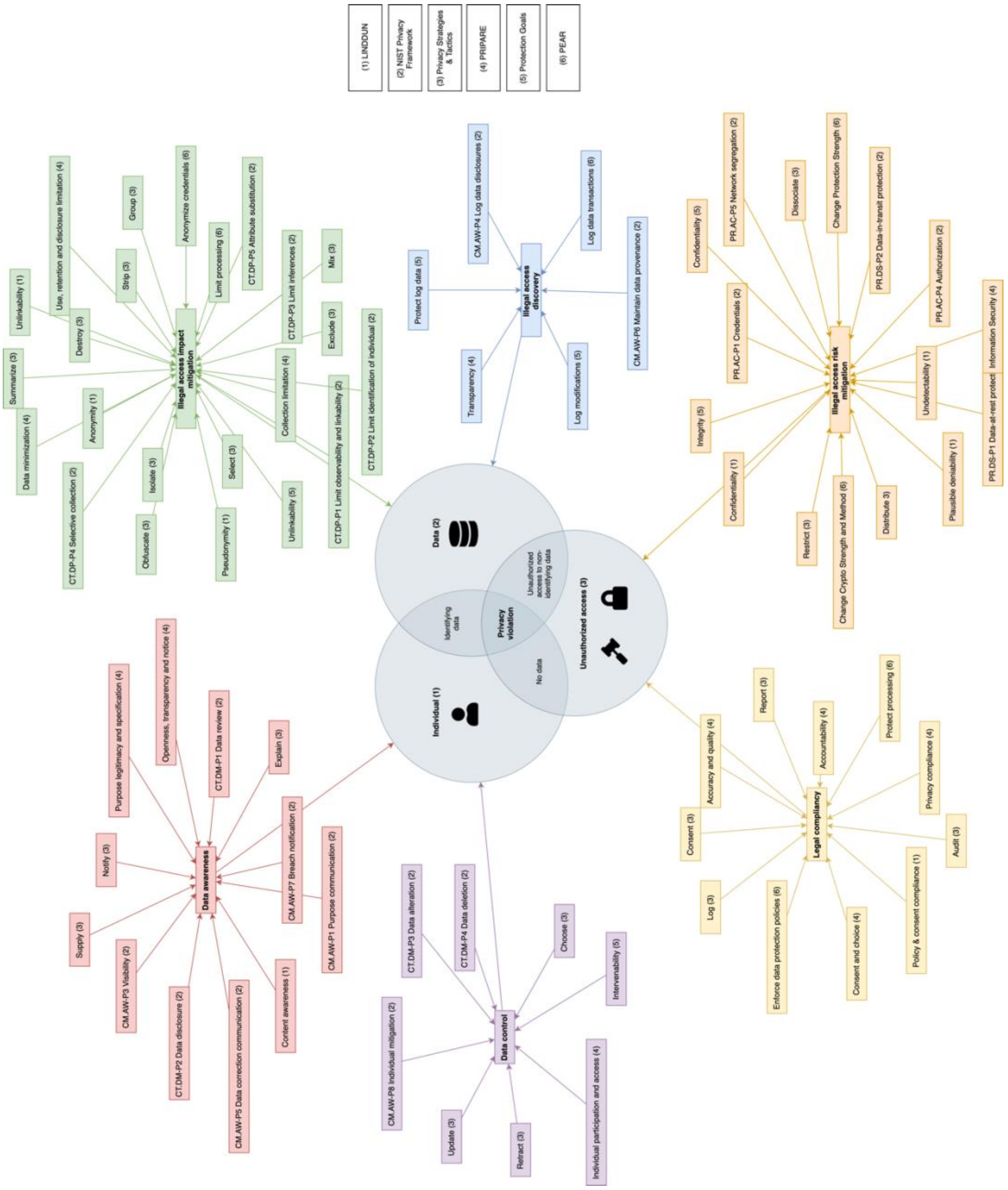

Analysis
Create functional description and high-level privacy analysis
Perform legal assessment
Prepare privacy and security plan
Create detailed privacy analysis
Operationalize privacy principles
Perform risk management
Design
Create Privacy Enhancing Architecture Design (PEAR)
Create Privacy Enhancing Detailed Design
Implementation
Implement Privacy
Verification
Check accountability
Perform Security & Privacy dynamic analysis
Perform Security & Privacy static analysis
Release
Create incidence response plan
Create system decomissioning plan
Perform Final Security & Privacy review
Publish PIA report
Maintenance
Execute Incident Response Plan
Perform Security & Privacy validations
Decomission
Execute Decomissioning Plan

Figure 19: Process-Deliverable Diagram of the PRIPARE study

7.3 Full Privacy Violation Diagram



(1) LINDDUN
(2) NIST Privacy Framework
(3) Privacy Strategies & Tactics
(4) PRIPARE
(5) Protection Goals
(6) PEAR
Data (2)
Individual (1)
Unauthorized access (3)
Identifying data
Privacy violation
Unauthorized access to non-identifying data
No data
Illegal access impact mitigation
Unlinkability (1)
Use, retention and disclosure limitation (4)
Group (3)
Strip (3)
Anonymize credentials (6)
Limit processing (6)
CT.DP-P5 Attribute substitution (2)
CT.DP-P3 Limit inferences (2)
Mix (3)
Exclude (3)
CT.DP-P2 Limit identification of individual (2)
Destroy (3)
Summarize (3)
Data minimization (4)
Anonymity (1)
CT.DP-P4 Selective collection (2)
Isolate (3)
Obfuscate (3)
Pseudonymity (1)
Select (3)
Unlinkability (5)
Collection limitation (4)
CT.DP-P1 Limit observability and linkability (2)
Illegal access discovery
CM.AW-P4 Log data disclosures (2)
Log data transactions (6)
CM.AW-P6 Maintain data provenance (2)
Protect log data (5)
Transparency (4)
Log modifications (5)
Illegal access risk mitigation
Confidentiality (5)
PR.AC-P5 Network segregation (2)
Dissociate (3)
Change Protection Strength (6)
PR.DS-P2 Data-in-transit protection (2)
PR.AC-P4 Authorization (2)
PR.AC-P1 Credentials (2)
Information Security (4)
PR.DS-P1 Data-at-rest protect
Undetectability (1)
Integrity (5)
Confidentiality (1)
Restrict (3)
Change Crypto Strength and Method (6)
Distribute 3)
Plausible deniability (1)
Legal compliancy
Accuracy and quality (4)
Report (3)
Accountability (4)
Protect processing (6)
Privacy compliance (4)
Consent (3)
Audit (3)
Log (3)
Enforce data protection policies (6)
Consent and choice (4)
Policy & consent compliance (1)
Data awareness
Purpose legitimacy and specification (4)
Openness, transparency and notice (4)
CT.DM-P1 Data review (2)
Explain (3)
CM.AW-P7 Breach notification (2)
CM.AW-P1 Purpose communication (2)
Notify (3)
Supply (3)
CM.AW-P3 Visibility (2)
CT.DM-P2 Data disclosure (2)
CM.AW-P5 Data correction communication (2)
Content awareness (1)
Data control
CT.DM-P3 Data alteration (2)
CT.DM-P4 Data deletion (2)
Choose (3)
Intervenability (5)
CM.AW-P8 Individual mitigation (2)
Update (3)
Retract (3)
Individual participation and access (4)

Figure 20: The privacy tactics from the studies LINDDUN, NIST, Strategies & Tactics, PRIPARE, Protection Goals and PEAR mapped on the PVD

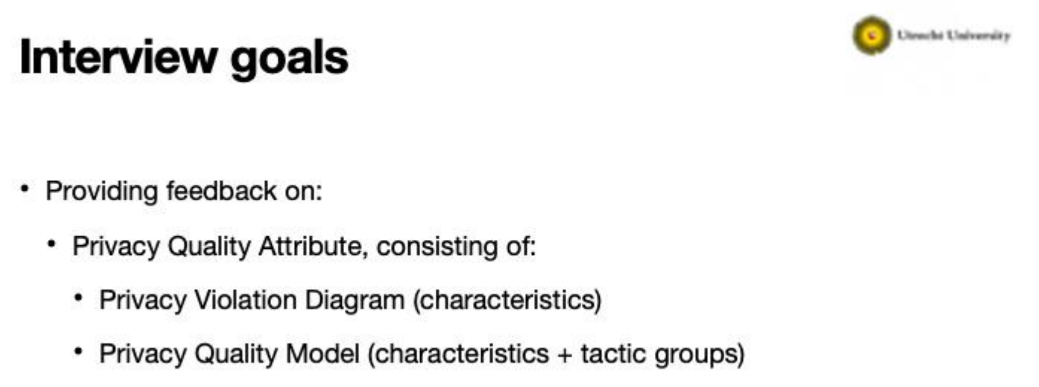
7.4 Expert interview slide deck

Addressing Privacy
in Software Architecture

Evaluation interview

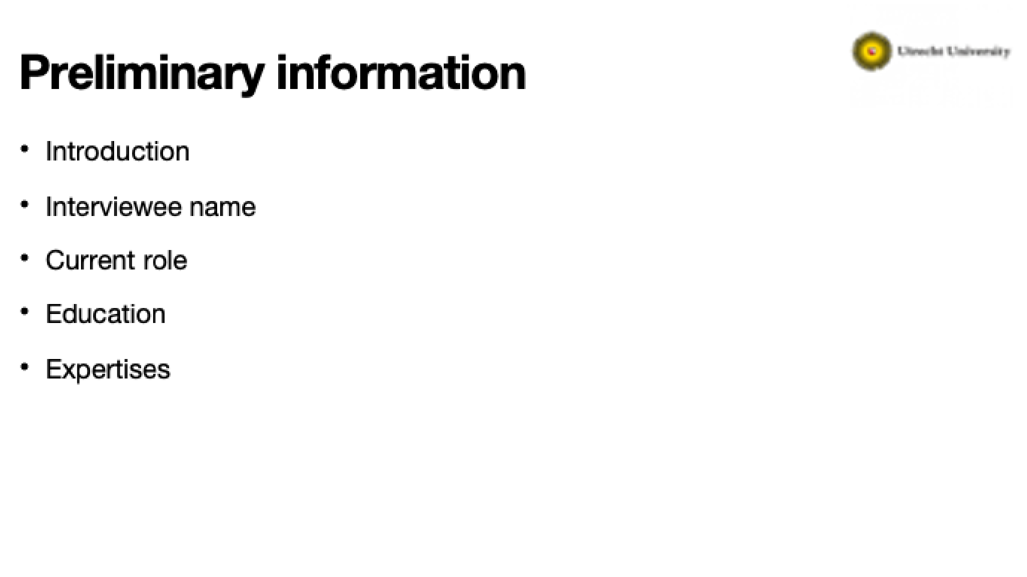
Koen Schellens
dr. Jan Martijn van der Werf
dr. Fabiano Dalpiaz

May, 2021



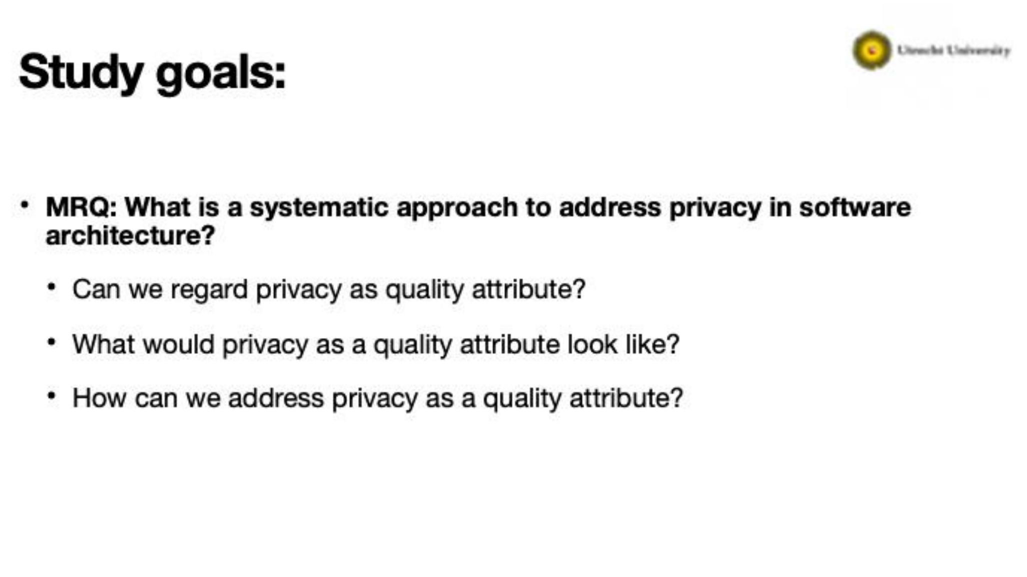
Interview goals

- Providing feedback on:
  - Privacy Quality Attribute, consisting of:
    - Privacy Violation Diagram (characteristics)
    - Privacy Quality Model (characteristics + tactic groups)



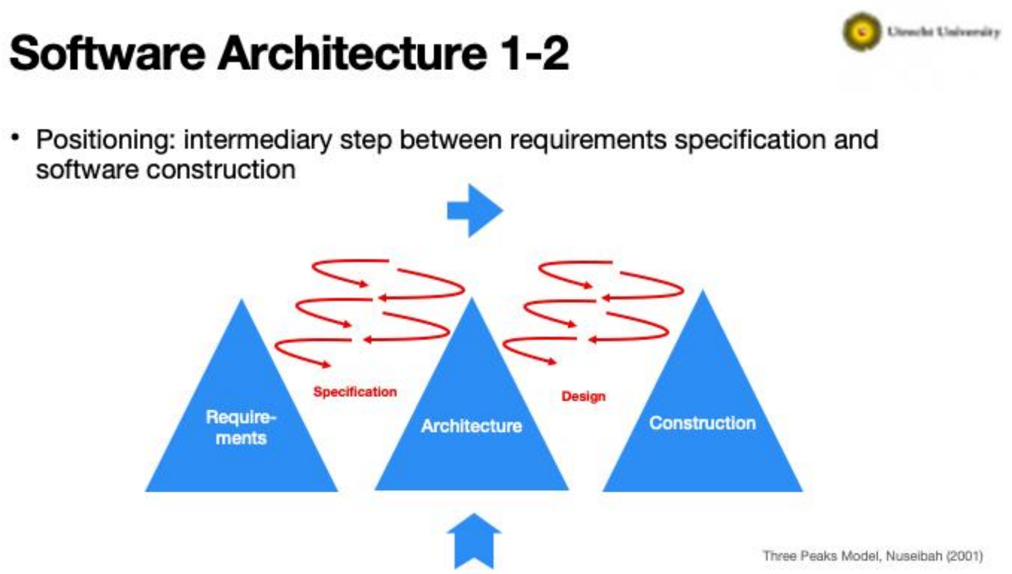
Preliminary information

- Introduction
- Interviewee name
- Current role
- Education
- Expertises



Study goals:

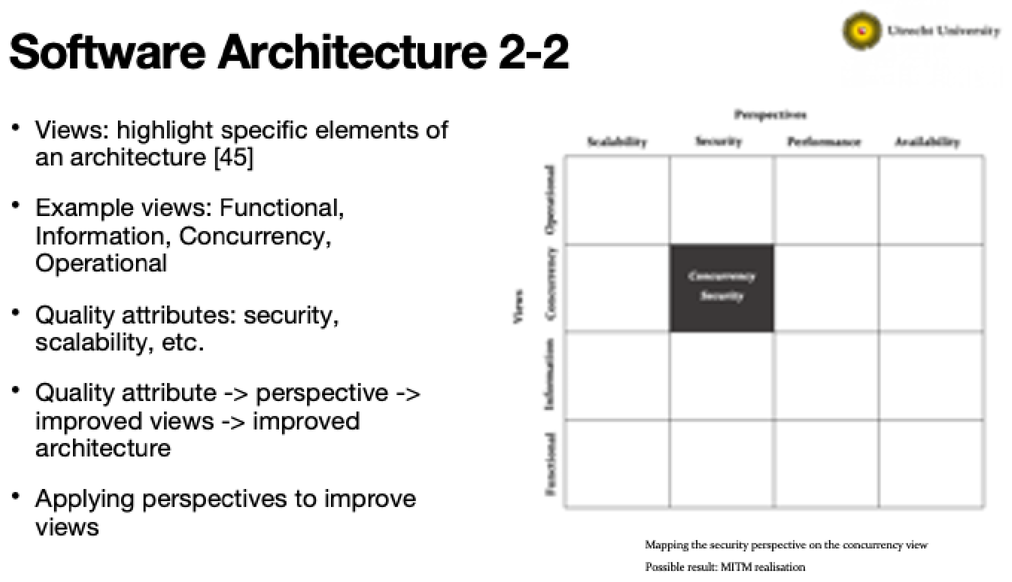
- **MRQ: What is a systematic approach to address privacy in software architecture?**
  - Can we regard privacy as quality attribute?
  - What would privacy as a quality attribute look like?
  - How can we address privacy as a quality attribute?



Software Architecture 1-2

- Positioning: intermediary step between requirements specification and software construction

Three Peaks Model, Nuseibah (2001)



Software Architecture 2-2

- Views: highlight specific elements of an architecture [45]
- Example views: Functional, Information, Concurrency, Operational
- Quality attributes: security, scalability, etc.
- Quality attribute -> perspective -> improved views -> improved architecture
- Applying perspectives to improve views

Views	Scalability	Security	Performance	Availability
Operational				
Concurrency		Concurrency Security		
Information				
Functional				

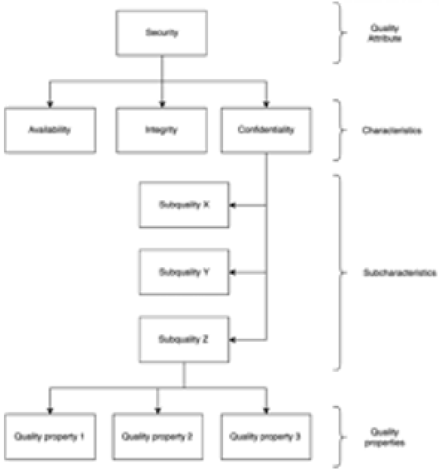
Perspectives

Mapping the security perspective on the concurrency view

Possible result: MITM realisation

Can we regard privacy as quality attribute?

- Literature: yes [47]
- Can we express it as a quality model (ISO 25010)?
- Characteristics, subcharacteristics, quality properties?

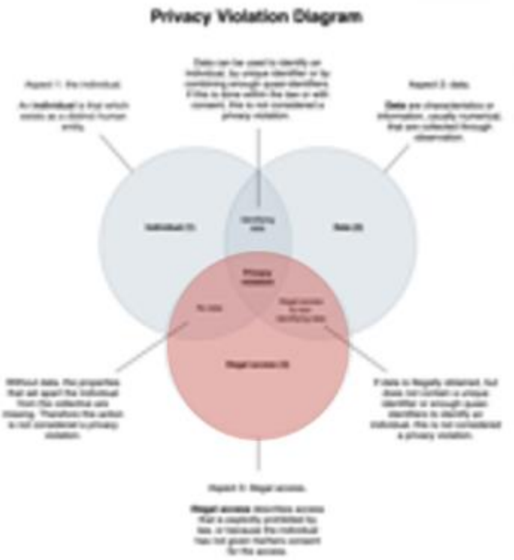


Privacy characteristic problems

- Studies propose characteristics, we are not sure
- Example: LINDDUN: unlinkability ("privacy property"). Improves privacy (tactic), but is not a property (characteristic).
- What vs. How (characteristic vs. tactic)
- Mix of characteristics and tactics
- Examples: privacy properties [58], privacy objectives [38], privacy design strategies [9], privacy principles [36], privacy protection goals [27] or privacy tactics [35]

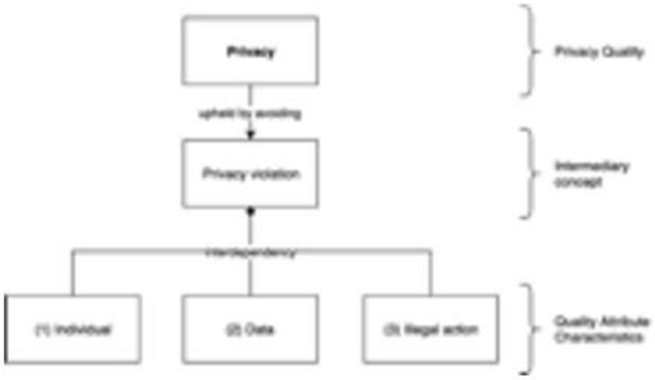
Privacy characteristics

- Westin, Solove, ISO 29100
- Privacy Violation, three key characteristics:
  1. Individual
  2. Data
  3. Illegal Access
- Interdependency


Privacy Violation Diagram
Aspect 1: the individual.
An individual is that which exists as a distinct human entity.
Data can be used to identify an individual, by unique identifier or by combining enough quasi-identifiers. If this is done within the law or with consent, this is not considered a privacy violation.
Aspect 2: data.
Data are characteristics or information, usually numerical, that are collected through observation.
Individual (1)
Identifying data
Data (2)
Privacy violation
No data
Illegal access to non identifying data
Illegal access (3)
Without data, the properties that set apart the individual from the collective are missing. Therefore the action is not considered a privacy violation.
If data is illegally obtained, but does not contain a unique identifier or enough quasi identifiers to identify an individual, this is not considered a privacy violation.
Aspect 3: Illegal access.
Illegal access describes access that is explicitly prohibited by law, or because the individual has not given his/hers consent for the access.

Privacy quality model

Utrecht University



Privacy tactics

Goal: select tactics to address characteristics

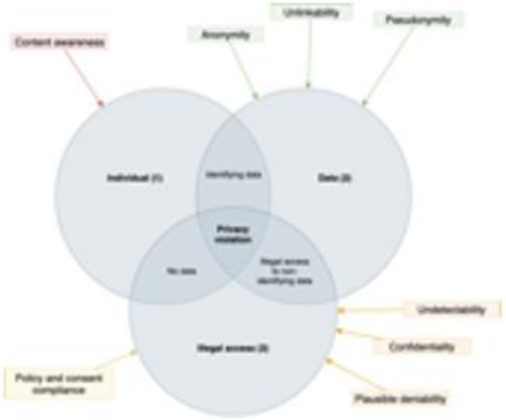
Analysed studies:

1. LINDDUN: A Privacy Threat Analysis Framework. Wuyts and Joosen, KU Leuven, 2014. [58]
2. NIST Privacy Framework. Hiller and Russell, NIST, 2017. [38]
3. A Critical Analysis of Privacy Design Strategies. Colesky et al., Radboud University, 2016. [9]
4. PRIPARE Methodology Handbook. Notario et al, Industry + Universities, EU and UK, 2015. [36]
5. Protection Goals for Privacy Engineering. Hansen et al, ULD Kiel Germany, 2015. [27]
6. PEAR: Privacy Enhancing ARchitectures. Kung, Trialog France, 2014. [35]

LINDDUN (example)

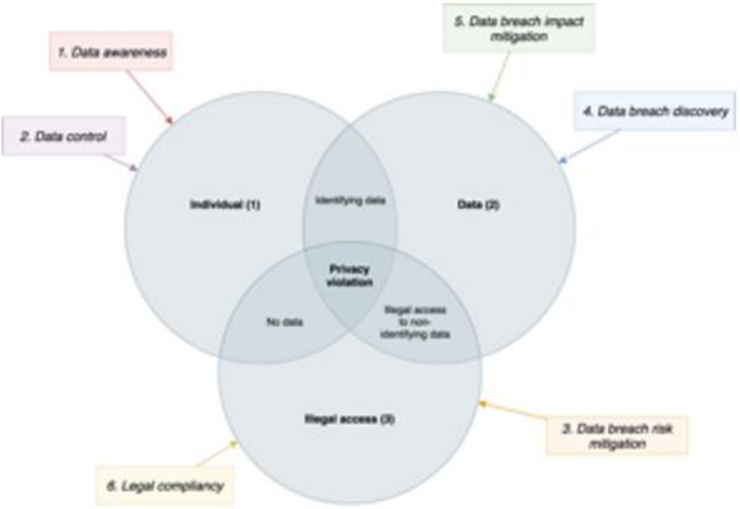
Steps per study:

- Extract tactics
- Select privacy tactics
- Group privacy tactics
- Map tactics on characteristics



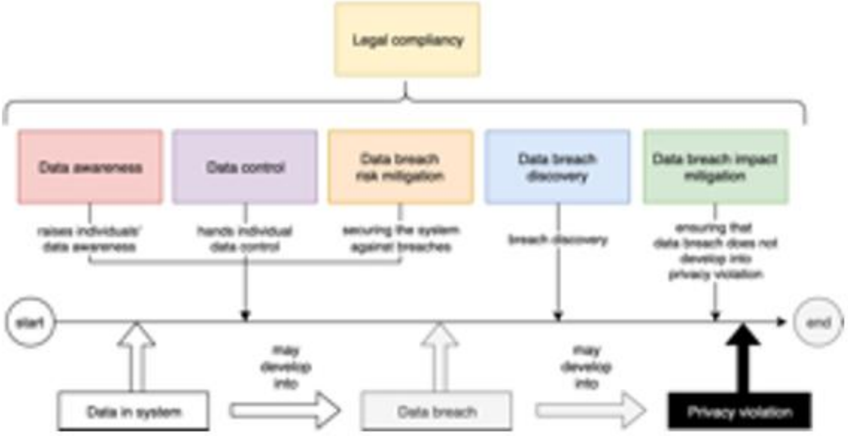
LINDDUN in the Privacy Violation Diagram

Tactic groups

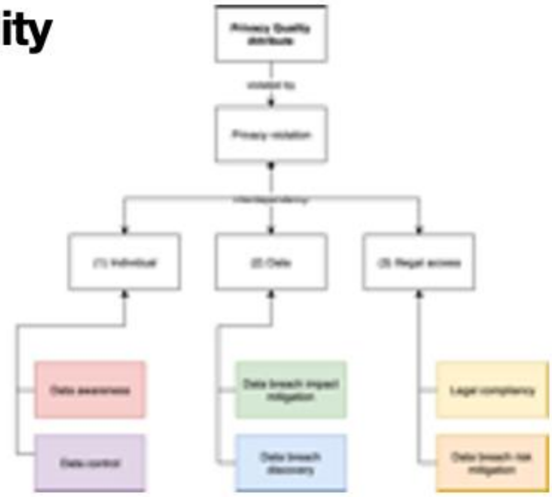


Tactic groups mapped on Privacy Violation Diagram

Privacy violation timeline


Legal compliancy
Data awareness
Data control
Data breach risk mitigation
Data breach discovery
Data breach impact mitigation
raises individuals' data awareness
hands individual data control
securing the system against breaches
breach discovery
ensuring that data breach does not develop into privacy violation
start
end
Data in system
may develop into
Data breach
may develop into
Privacy violation

Privacy Quality Model

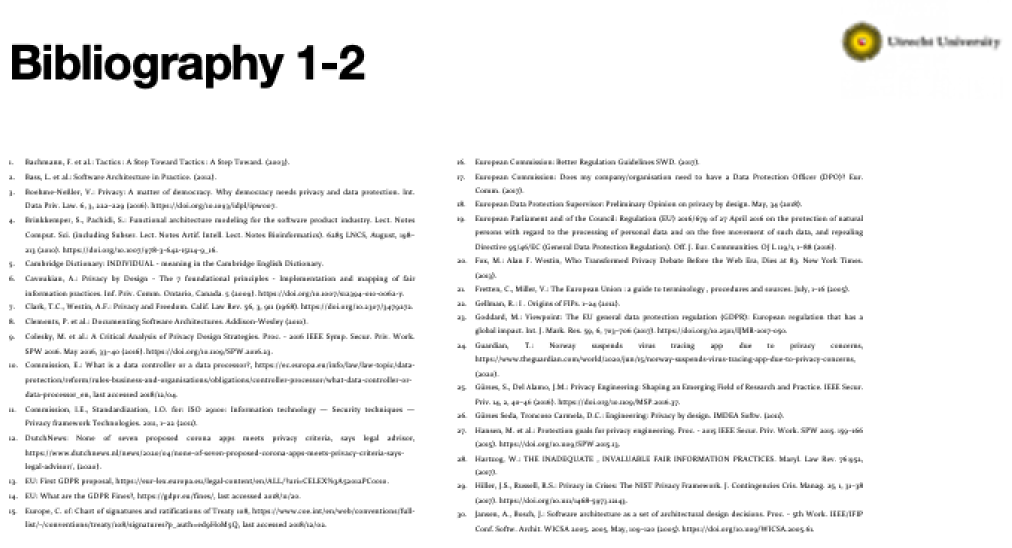


Privacy Quality Model + Tactic Groups

Correctness and completeness

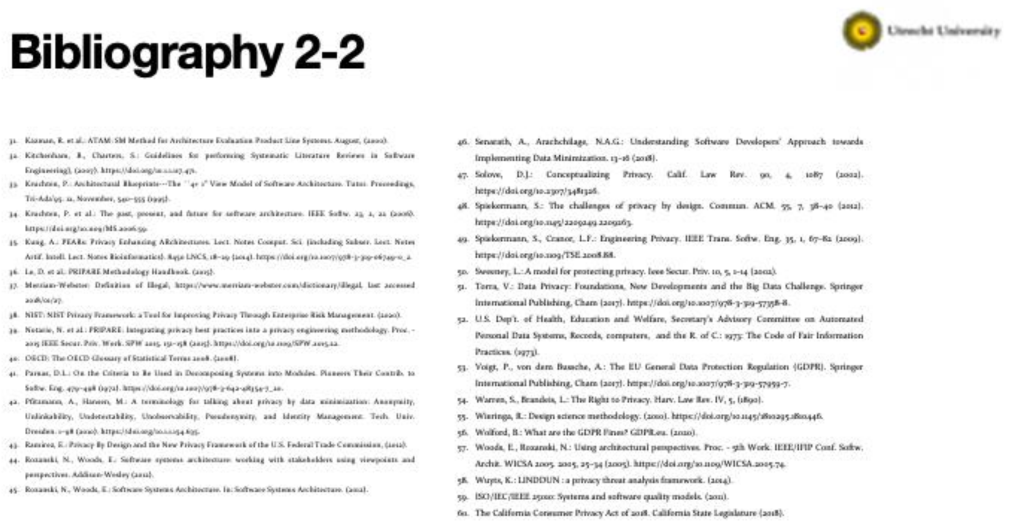
Utrecht University

- Privacy Violation Diagram
- Privacy Quality Model (characteristics)
- Privacy Tactic Groups (tactics)



Bibliography 1-2

1. Bachmann, F. et al.: Tactics : A Step Toward Tactics : A Step Toward. (2003).
2. Bass, L. et al.: Software Architecture in Practice. (2012).
3. Boehme-Neßler, V.: Privacy: A matter of democracy. Why democracy needs privacy and data protection. Int. Data Priv. Law. 6, 3, 222–229 (2016). https://doi.org/10.1093/idpl/ipw007.
4. Brinkkemper, S., Pachidi, S.: Functional architecture modeling for the software product industry. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 6285 LNCS, August, 198–213 (2010). https://doi.org/10.1007/978-3-642-15114-9_16.
5. Cambridge Dictionary: INDIVIDUAL - meaning in the Cambridge English Dictionary.
6. Cavoukian, A.: Privacy by Design - The 7 foundational principles - Implementation and mapping of fair information practices. Inf. Priv. Comm. Ontario, Canada. 5 (2009). https://doi.org/10.1007/s12394-010-0062-y.
7. Clark, T.C., Westin, A.F.: Privacy and Freedom. Calif. Law Rev. 56, 3, 911 (1968). https://doi.org/10.2307/3479432.
8. Clements, P. et al.: Documenting Software Architectures. Addison-Wesley (2010).
9. Colesky, M. et al.: A Critical Analysis of Privacy Design Strategies. Proc. - 2016 IEEE Symp. Secur. Priv. Work. SPW 2016. May 2016, 33–40 (2016). https://doi.org/10.1109/SPW.2016.23.
10. Commission, E.: What is a data controller or a data processor?, https://ec.europa.eu/info/law/law-topic/data-protection/reform/rules-business-and-organisations/obligations/controller-processor/what-data-controller-or-data-processor_en, last accessed 2018/12/04.
11. Commission, I.E., Standardization, I.O. for: ISO 29100: Information technology — Security techniques — Privacy framework Technologies. 2011, 1–22 (2011).
12. DutchNews: None of seven proposed corona apps meets privacy criteria, says legal advisor, https://www.dutchnews.nl/news/2020/04/none-of-seven-proposed-corona-apps-meets-privacy-criteria-says-legal-advisor/, (2020).
13. EU: First GDPR proposal, https://eur-lex.europa.eu/legal-content/en/ALL/?uri=CELEX%3A52012PC0011.
14. EU: What are the GDPR Fines?, https://gdpr.eu/fines/, last accessed 2018/11/20.
15. Europe, C. of: Chart of signatures and ratifications of Treaty 108, https://www.coe.int/en/web/conventions/full-list/-/conventions/treaty/108/signatures?p_auth=zdgHoM5Q, last accessed 2018/12/02.
16. European Commission: Better Regulation Guidelines SWD. (2017).
17. European Commission: Does my company/organisation need to have a Data Protection Officer (DPO)? Eur. Comm. (2017).
18. European Data Protection Supervisor: Preliminary Opinion on privacy by design. May, 34 (2018).
19. European Parliament and of the Council: Regulation (EU) 2016/679 of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). Off. J. Eur. Communities. OJ L 119/1, 1–88 (2016).
20. Fox, M.: Alan F. Westin, Who Transformed Privacy Debate Before the Web Era, Dies at 83. New York Times. (2013).
21. Fretten, C., Miller, V.: The European Union : a guide to terminology , procedures and sources. July, 1–16 (2005).
22. Gellman, R. : I . Origins of FIPs. 1–24 (2012).
23. Goddard, M.: Viewpoint: The EU general data protection regulation (GDPR): European regulation that has a global impact. Int. J. Mark. Res. 59, 6, 703–706 (2017). https://doi.org/10.2501/IJMR-2017-050.
24. Guardian, T.: Norway suspends virus tracing app due to privacy concerns, https://www.theguardian.com/world/2020/jun/15/norway-suspends-virus-tracing-app-due-to-privacy-concerns, (2020).
25. Gürses, S., Del Alamo, J.M.: Privacy Engineering: Shaping an Emerging Field of Research and Practice. IEEE Secur. Priv. 14, 2, 40–46 (2016). https://doi.org/10.1109/MSP.2016.37.
26. Gürses Seda, Troncoso Carmela, D.C.: Engineering: Privacy by design. IMDEA Softw. (2011).
27. Hansen, M. et al.: Protection goals for privacy engineering. Proc. - 2015 IEEE Secur. Priv. Work. SPW 2015. 159–166 (2015). https://doi.org/10.1109/SPW.2015.13.
28. Hartzog, W.: THE INADEQUATE , INVALUABLE FAIR INFORMATION PRACTICES. Maryl. Law Rev. 76 952, (2017).
29. Hiller, J.S., Russell, R.S.: Privacy in Crises: The NIST Privacy Framework. J. Contingencies Cris. Manag. 25, 1, 31–38 (2017). https://doi.org/10.1111/1468-5973.12143.
30. Jansen, A., Bosch, J.: Software architecture as a set of architectural design decisions. Proc. - 5th Work. IEEE/IFIP Conf. Softw. Archit. WICSA 2005. 2005, May, 109–120 (2005). https://doi.org/10.1109/WICSA.2005.61.



Bibliography 2-2

31. Kazman, R. et al.: ATAM: SM Method for Architecture Evaluation Product Line Systems. August, (2000).
32. Kitchenham, B., Charters, S.: Guidelines for performing Systematic Literature Reviews in Software Engineering}, (2007). https://doi.org/10.1.1.117.471.
33. Kruchten, P.: Architectural Blueprints—The "4+1" View Model of Software Architecture. Tutor. Proceedings, Tri-Ada'95. 12, November, 540–555 (1995).
34. Kruchten, P. et al.: The past, present, and future for software architecture. IEEE Softw. 23, 2, 22 (2006). https://doi.org/10.1109/MS.2006.59.
35. Kung, A.: PEARs: Privacy Enhancing ARchitectures. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 8450 LNCS, 18–29 (2014). https://doi.org/10.1007/978-3-319-06749-0_2.
36. Le, D. et al.: PRIPARE Methodology Handbook. (2015).
37. Merriam-Webster: Definition of Illegal, https://www.merriam-webster.com/dictionary/illegal, last accessed 2018/01/27.
38. NIST: NIST Privacy Framework: a Tool for Improving Privacy Through Enterprise Risk Management. (2020).
39. Notario, N. et al.: PRIPARE: Integrating privacy best practices into a privacy engineering methodology. Proc. - 2015 IEEE Secur. Priv. Work. SPW 2015. 151–158 (2015). https://doi.org/10.1109/SPW.2015.22.
40. OECD: The OECD Glossary of Statistical Terms 2008. (2008).
41. Parnas, D.L.: On the Criteria to Be Used in Decomposing Systems into Modules. Pioneers Their Contrib. to Softw. Eng. 479–498 (1972). https://doi.org/10.1007/978-3-642-48354-7_20.
42. Pfitzmann, A., Hansen, M.: A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management. Tech. Univ. Dresden. 1–98 (2010). https://doi.org/10.1.1.154.635.
43. Ramirez, E.: Privacy By Design and the New Privacy Framework of the U.S. Federal Trade Commission, (2012).
44. Rozanski, N., Woods, E.: Software systems architecture: working with stakeholders using viewpoints and perspectives. Addison-Wesley (2012).
45. Rozanski, N., Woods, E.: Software Systems Architecture. In: Software Systems Architecture. (2012).
46. Senarath, A., Arachchilage, N.A.G.: Understanding Software Developers' Approach towards Implementing Data Minimization. 13–18 (2018).
47. Solove, D.J.: Conceptualizing Privacy. Calif. Law Rev. 90, 4, 1087 (2002). https://doi.org/10.2307/3481326.
48. Spiekermann, S.: The challenges of privacy by design. Commun. ACM. 55, 7, 38–40 (2012). https://doi.org/10.1145/2209249.2209263.
49. Spiekermann, S., Cranor, L.F.: Engineering Privacy. IEEE Trans. Softw. Eng. 35, 1, 67–82 (2009). https://doi.org/10.1109/TSE.2008.88.
50. Sweeney, L.: A model for protecting privacy. Ieee Secur. Priv. 10, 5, 1–14 (2002).
51. Torra, V.: Data Privacy: Foundations, New Developments and the Big Data Challenge. Springer International Publishing, Cham (2017). https://doi.org/10.1007/978-3-319-57358-8.
52. U.S. Dep't. of Health, Education and Welfare, Secretary's Advisory Committee on Automated Personal Data Systems, Records, computers, and the R. of C.: 1973: The Code of Fair Information Practices. (1973).
53. Voigt, P., von dem Bussche, A.: The EU General Data Protection Regulation (GDPR). Springer International Publishing, Cham (2017). https://doi.org/10.1007/978-3-319-57959-7.
54. Warren, S., Brandeis, L.: The Right to Privacy. Harv. Law Rev. IV, 5, (1890).
55. Wieringa, R.: Design science methodology. (2010). https://doi.org/10.1145/1810295.1810446.
56. Wolford, B.: What are the GDPR Fines? GDPR.eu. (2020).
57. Woods, E., Rozanski, N.: Using architectural perspectives. Proc. - 5th Work. IEEE/IFIP Conf. Softw. Archit. WICSA 2005. 2005, 25–34 (2005). https://doi.org/10.1109/WICSA.2005.74.
58. Wuyts, K.: LINDDUN : a privacy threat analysis framework. (2014).
59. ISO/IEC/IEEE 25010: Systems and software quality models. (2011).
60. The California Consumer Privacy Act of 2018. California State Legislature (2018).