Model-Based Privacy by Design

by
Amirshayan Ahmadian, M.Sc.

Approved Dissertation thesis for the partial fulfilment of the requirements for a
Doctor of Natural Sciences (Dr. rer. nat.)
Fachbereich 4: Informatik
Universität Koblenz-Landau

Chair of PhD Board: Prof. Dr. Maria A. Wimmer
Chair of PhD Commission: Prof. Dr. Harald F.O. Von Korflesch
Examiner and Supervisor: Prof. Dr. Jan Jürjens
Further Examiner: Prof. Dr. Patrick Delfmann
Co-Supervisor: Dr. Daniel Strüber

Date of the doctoral viva: January 22, 2020

Dedicated to my beloved parents.

Abstract

Nowadays, almost any IT system involves personal data processing. In such systems, many privacy risks arise when privacy concerns are not properly addressed from the early phases of the system design. The General Data Protection Regulation (GDPR) prescribes the *Privacy by Design* (*PbD*) principle. As its core, *PbD* obliges protecting personal data from the onset of the system development, by effectively integrating appropriate privacy controls into the design. To operationalize the concept of *PbD*, a set of challenges emerges: First, we need a basis to define privacy concerns. Without such a basis, we are not able to verify whether personal data processing is authorized. Second, we need to identify where precisely in a system, the controls have to be applied. This calls for system analysis concerning privacy concerns. Third, with a view to selecting and integrating appropriate controls, based on the results of system analysis, a mechanism to identify the privacy risks is required. Mitigating privacy risks is at the core of the *PbD* principle. Fourth, choosing and integrating appropriate controls into a system are complex tasks that besides risks, have to consider potential interrelations among privacy controls and the costs of the controls.

This thesis introduces a model-based privacy by design methodology to handle the above challenges. Our methodology relies on a precise definition of privacy concerns and comprises three sub-methodologies: *model-based privacy analysis*, *model-based privacy impact assessment* and *privacy-enhanced system design modeling*. First, we introduce a definition of privacy preferences, which provides a basis to specify privacy concerns and to verify whether personal data processing is authorized. Second, we present a model-based methodology to analyze a system model. The results of this analysis denote a set of privacy design violations. Third, taking into account the results of privacy analysis, we introduce a model-based privacy impact assessment methodology to identify concrete privacy risks in a system model. Fourth, concerning the risks, and taking into account the interrelations and the costs of the controls, we propose a methodology to select appropriate controls and integrate them into a system design. Using various practical case studies, we evaluate our concepts, showing a promising outlook on the applicability of our methodology in real-world settings.

Zusammenfassung

In IT-Systemen treten viele Datenschutzrisiken auf, wenn Datenschutzbedenken in den frühen Phasen des Entwicklungsprozesses nicht angemessen berücksichtigt werden. Die Datenschutz-Grundverordnung (DSGVO) schreibt das Prinzip des *Datenschutz durch Technikgestaltung* (*PbD*) vor. *PbD* erfordert den Schutz personenbezogener Daten von Beginn des Entwicklungsprozesses an, durch das frühzeitige Integrieren geeigneter Maßnahmen. Bei der Realisierung von *PbD* ergeben sich nachfolgende Herausforderungen: Erstens benötigen wir eine präzise Definition von Datenschutzbedenken. Zweitens müssen wir herausfinden, wo genau in einem System die Maßnahmen angewendet werden müssen. Drittens ist zur Auswahl geeigneter Maßnahmen, ein Mechanismus zur Ermittlung der Datenschutzrisiken erforderlich. Viertens müssen bei der Auswahl und Integration geeigneter Maßnahmen, neben den Risiken, die Abhängigkeiten zwischen Maßnahmen und die Kosten der Maßnahmen berücksichtigt werden.

Diese Dissertation führt eine modellbasierte Methodik ein, um die oben genannten Herausforderungen zu bewältigen und *PbD* zu operationalisieren. Unsere Methodik basiert auf einer präzisen Definition von Datenschutzbedenken und umfasst drei Untermethodiken: *modellbasierte Datenschutzanalyse*, *modellbasierte Datenschutz-Folgenabschätzung* und *datenschutzfreundliche Systemmodellierung*. Zunächst führen wir eine Definition für Datenschutzpräferenzen ein, anhand derer die Datenschutzbedenken präzisiert werden können und überprüft werden kann, ob die Verarbeitung personenbezogener Daten autorisiert ist. Zweitens präsentieren wir eine modellbasierte Methodik zur Analyse eines Systemmodells. Die Ergebnisse dieser Analyse ergeben die Menge der Verstöße gegen die Datenschutzpräferenzen in einem Systemmodell. Drittens führen wir eine modellbasierte Methode zur Datenschutz-folgenabschätzung ein, um konkrete Datenschutzrisiken in einem Systemmodell zu identifizieren. Viertens schlagen wir in Bezug auf die Risiken, Abhängigkeiten zwischen Maßnahmen und Kosten der Maßnahmen, eine Methodik vor, um geeignete Maßnahmen auszuwählen und in ein Systemdesign zu integrieren. In einer Reihe von realistischen Fallstudien bewerten wir unsere Konzepte und geben einen vielversprechenden Ausblick auf die Anwendbarkeit unserer Methodik in der Praxis.

Contents

Abstract v

Zusammenfassung vii

Abbreviations xxi

Acknowledgements xxiii

1 Introduction 1

1.1 Challenges and Research Directions 4
1.1.1 Privacy Preferences 5
1.1.2 Privacy Analysis 5
1.1.3 Privacy Impact Assessment 6
1.1.4 Privacy Enhancement 6
1.2 Contributions 7
1.3 Methodology 9
1.4 Thesis Outline 11
1.5 How to Read this PhD Thesis 12
1.6 Preliminary Publications 13

2 Model-Based Privacy by Design: An Overview of the Methodology 15

2.1 The Common Terms in this Thesis 15
2.2 Running Example 16
2.3 Walk-Through: Model-Based Privacy by Design 20
2.3.1 Privacy Preferences 20
2.3.2 Model-Based Privacy Analysis 22
2.3.3 Model-Based Privacy Impact Assessment 22
2.3.4 Privacy-Enhanced System Design Modeling 23

3 Privacy Preferences: A Foundation 25

3.1 Introduction 26

3.2 Background 27
3.2.1 The Four Key Elements of Privacy 27
3.2.2 A Background on the Theory of Sets and Lattices 28
3.2.3 Privacy Level Agreements 30
3.3 Privacy Preferences 31
3.4 Formalized Privacy Level Agreements 43
3.4.1 A Brief Description of the Differences Between the GDPR and Directive 95/46/EC 43
3.4.2 The PLA Metamodel 44
3.5 Discussion and Limitations 46
3.5.1 Revisiting the Research Questions 46
3.5.2 Limitations 47
3.6 Related Work 48
3.7 Preliminary Conclusion 51

4 Model-Based Privacy Analysis 53

4.1 Introduction 54
4.2 Background 56
4.2.1 Unified Modeling Language (UML) 56
4.2.1.1 Class Diagram 57
4.2.1.2 Activity Diagram 57
4.2.1.3 Deployment Diagram 58
4.2.1.4 State Machines 59
4.2.2 UML Profile 59
4.2.3 Model-Based Security Analysis Using UMLsec 60
4.2.3.1 Secure Links 60
4.2.3.2 Secure Dependency 62
4.3 Model-Based Privacy Analysis in Industrial Ecosystems 64
4.3.1 The Modular Privacy Analysis 64
4.3.2 Privacy Analysis Based on the Four Fundamental Privacy Elements 66
4.3.3 UML Privacy Extension 68
4.3.3.1 Privacy Profile 69
4.3.3.2 *rabac* Profile 71
4.3.4 The Privacy Checks 73
4.3.5 Applying the Privacy Checks: Example 80
4.4 Case Studies and a Survey 82
4.4.1 Case Studies 83
4.4.2 Investigating the Required Support to Carry out the Proposed Methodology 85
4.4.2.1 A Survey on System Modeling within the VisiOn Project 85

4.4.2.2 Our Observation and Conclusion 87
4.5 Discussion and Limitations 89
4.6 Related Work 91
4.7 Preliminary Conclusion 93

5 Supporting Privacy Impact Assessment by Model-Based Privacy Analysis 95

5.1 Introduction 96
5.2 Background 97
5.2.1 Risk, Threat, and Risk Analysis 97
5.2.2 Privacy Impact Assessment 98
5.2.3 Privacy Targets 99
5.2.4 Privacy Threats 99
5.3 Model-Based Privacy Impact Assessment 100
5.3.1 Systematic Specification of System and its Privacy-Critical Parts 100
5.3.2 Model-Based Privacy and Security Analysis 102
5.3.3 Identification of Harmful Activities and Threats 103
5.3.4 Impact Assessment 106
5.3.5 Identification of Appropriate Controls 113
5.3.6 Privacy Impact Assessment Report 114
5.4 Case Studies and Evaluation 115
5.4.1 Case Studies 115
5.4.2 Comparative Evaluation 116
5.5 Discussion and Limitations 118
5.6 Related Work 120
5.7 Preliminary Conclusion 123

6 Privacy-Enhanced System Design Modeling Based on Privacy Features 125

6.1 Introduction 126
6.2 Background 128
6.2.1 Privacy Design Strategies, Patterns, and Privacy-Enhancing Technologies 128
6.2.2 Function Point Analysis (FPA) 128
6.2.3 Reusable Aspect Models (RAMs) 129
6.3 Running Example 130
6.4 Privacy-Enhanced System Design Modeling 132
6.4.1 The Privacy Design Strategies Feature Model 133
6.4.2 Model-Based Cost Estimation 136
6.4.3 Model-Based Privacy Enhancement 138
6.4.3.1 UML Profile for Privacy Enhancement. 138
6.4.3.2 Using and Extending RAMs for Privacy by Design. . 141
6.5 Case Studies and Evaluation 143
6.6 Discussion and Limitations 144

6.7 Related Work 146
6.8 Preliminary Conclusion 147

7 Tool Support 149

7.1 Introduction 149
7.2 Model-Based Privacy Analysis with CARiSMA in the Context of the VPP 150
7.2.1 Overview and New Features 152
7.2.2 Security and Privacy Analysis within the VisiOn Privacy Platform 153
7.2.2.1 The Architecture of VPP 153
7.2.2.2 System Model Analysis Using CARiSMA 155
7.2.2.3 RABAC 157
7.2.3 Privacy Level Agreements within the VisiOn Privacy Platform 161
7.2.4 Implementation and Availability 164
7.2.5 Related Work 166
7.2.6 Preliminary Conclusion 167
7.3 Pattern-Based Risk Analysis with the ClouDAT Framework 167
7.3.1 The Overview of the ClouDAT Risk Analysis Process 169
7.3.2 The CLouDAT's Risk Analysis Metamodel 171
7.3.3 The ClouDAT's Risk Treatment Method 173
7.3.3.1 The Structure of the Control List 173
7.3.3.2 Risk Treatment Process 175
7.3.4 Supporting a Privacy Impact Assessment Using the ClouDAT Framework 177

8 Extending Model-Based Privacy Analysis for the IDS: A Case Study 181

8.1 Introduction 182
8.2 Addressing Privacy in the IDS 183
8.3 Model-based Privacy Analysis for the IDS 185
8.3.1 Privacy Preferences 186
8.3.2 Privacy Analysis 187
8.4 Discussion and Results 190
8.4.1 A Validation of the Model-Based Privacy Analysis Concerning Privacy Targets 190
8.4.2 Supporting Security and Privacy Analysis by Model-Based Risk Analysis in the IDS 191
8.5 Related Work 194
8.6 Preliminary Conclusion 195

9 Conclusions, Limitations and Outlook 197

9.1 Conclusion 197

9.2 Assumptions and Limitations . . . 200
9.3 Outlook . . . 202
9.3.1 Privacy Analysis at Runtime . . . 202
9.3.2 Model-Based Discrimination Analysis . . . 202
9.3.3 Investigating the Means to Support Performing a Model-Based Privacy Analysis . . . 203
9.3.4 Refined Cost Estimation . . . 203

A Glossary 205

B A Comparison Between the GDPR and the Directive 95/46/EC 209

C The Updated PLA Outline 211

D Solove's Harmful Activities, the Privacy Targets, and the Analysis Means 213

E A Mapping Between NIST Privacy Controls and the Privacy Targets 215

F Privacy Feature Tables 217

G A Walk Through of CARiSMA's Screencast 225
G.1 Tool Installation . . . 225
G.2 Help Content . . . 226
G.3 The Script of the Screencast . . . 226

H Lebenslauf 229

Bibliography 231

List of Figures

1.1 The foundation of our *PbD* methodology focusing only on our contributions in this thesis 7
1.2 The outline of the thesis, showing how to read this thesis 13

2.1 An illustration of the industrial ecosystem where MoA (*Municipality of Athens*) sends personal data to other service providers 17
2.2 The overall workflow of the model-based privacy by design 21

3.1 The highlighted section denotes how Chapter 3 contributes to the overall workflow. 25
3.2 A sample *purpose lattice* 33
3.3 A sample *visibility lattice* 34
3.4 A *chain* showing the four granularity levels 35
3.5 A sample *retention lattice* 36
3.6 The left lattice illustrates the visibility lattice $(VL, \leq)$. The right lattice shows the powerset of V $(\mathcal{P}(V))$. 37
3.7 An example of the privacy preferences of *SSN*. The dashed lines indicate the privacy preferences (authorized purposes, subjects, granularity levels and retention conditions). 38
3.8 The effective privacy preferences of *SSN*. The dashed lines indicate the privacy preferences (authorized purposes, subjects, granularity levels, and retention conditions). 42
3.9 The PLA metamodel 45

4.1 The highlighted section denotes how Chapter 4 contributes to the overall workflow. 53
4.2 Design model excerpt (deployment diagram) annotated with «`secure links`» 62
4.3 Design model excerpt (class diagram) annotated with «`secure dependency`» 63
4.4 Model-based privacy analysis by exploiting PLAs 64
4.5 A sample illustration of an industrial ecosystem, which entails the *Municipality of Athens* (*MoA*). 65
4.6 Design model excerpt (*issue birth certificate* activity), which highlights the need to perform a privacy analysis. 67

4.7 Model of the privacy profile 71
4.8 Design model excerpt annotated with the privacy and *rabac* profiles . 72
4.9 Design model excerpt with highlighted annotations 80
4.10 The privacy preferences of *SSN* (see Section 3.3). 81
4.11 An excerpt of the survey results showing the expertise of our industry partners in system modeling 86

5.1 The highlighted section denotes how Chapter 5 contributes to the overall workflow. 95
5.2 *Privacy impact assessment* supported by model-based privacy analysis 101
5.3 The figure shows Step 2 and 3 of the *PIA* methodology. 102
5.4 Design model excerpt (*issue birth certificate* activity). 104
5.5 The privacy preferences of *SSN* (see Section 3.3). 105

6.1 The highlighted section denotes how Chapter 6 contributes to the overall workflow. 125
6.2 Design model excerpt (*issue birth certificate* activity). 130
6.3 The privacy preferences of *SSN* (see Section 3.3). 131
6.4 The workflow of the methodology proposed in this chapter to enhance a system model with privacy controls 132
6.5 An excerpt of the feature model including privacy design strategies, sub-strategies, privacy patterns, and PETs 135
6.6 The metamodel demonstrating the underlying concepts of the privacy enhancement 140
6.7 An excerpt of the privacy-enhanced system model 141
6.8 A screenshot showing an excerpt of a feature model's configuration in *FeatureIDE* 142
6.9 The *Authorization* aspect including a dataflow view 143

7.1 Model-based security and privacy analysis 152
7.2 An excerpt from the architecture of the VisiOn privacy platform (VPP).153
7.3 A document excerpt generated by STS, listing the security and privacy requirements 154
7.4 A screenshot excerpt of CARiSMA demonstrating the pre-analysis . 155
7.5 CARiSMA offers several options after running a pre-analysis. 156
7.6 A screenshot excerpt of CARiSMA showing a help report excerpt generated by CARiSMA after running the pre-analysis 156
7.7 A screenshot excerpt of CARiSMA showing an automatically generated CARiSMA analysis after running a pre-analysis 157
7.8 A class diagram excerpt of the MoA system model 159
7.9 A properties view excerpt showing the profile defined for the *issueBirthCertificate* operation of the *MACS* class 159
7.10 A screenshot showing the RABAC check 160

7.11 The RABAC pop up menu to insert required input (including a user and an attribute) for a RABAC analysis 160
7.12 An excerpt of the analysis result following performing a RABAC analysis 161
7.13 A PLA instance [68] generated by the VPP in the context of the VisiOn project with respect to the MoA case study. 163
7.14 The architecture of CARiSMA extended by the relevant VisiOn components 165
7.15 The overview of the pattern-based risk analysis process introduced in [12] (Inst. stands for the instantiate). 169
7.16 Excerpt of the risk analysis metamodel [195] 171
7.17 Excerpt of the selectable text metamodel [195]. The metamodel demonstrates the element of a requirement or a threat pattern. 172
7.18 An example of a security requirement pattern 172
7.19 An example of a threat pattern 172
7.20 A screenshot, showing a template to create a security requirement pattern using the ClouDAT framework [195] 176
7.21 An excerpt of the ClouDAT control list 178
7.22 A privacy target pattern (similar to a security requirement pattern demonstrated in Figure 7.18). 179

8.1 An illustration of the IDS system layer, including three organizations 185
8.2 An illustration of the IDS system layer, including PLAs 186
8.3 Design model excerpt annotated with privacy profile 188
8.4 Design model excerpt 189

D.1 Solove's harmful activities, the privacy targets, and the analysis means.214

List of Tables

4.1 The UMLsec default adversary pattern 61
4.2 *Privacy* profile 69
4.3 *rabac* profile 73
4.4 Information on the three case studies 84

5.1 The mapping between harmful activities and the privacy/security checks 103
5.2 The relation between the privacy targets and the privacy/security analysis 109
5.3 Personal Data Category Values (*PDCV*s) 110
5.4 Impact Values 112
5.5 The categorization of the IA Scores 113
5.6 The enhanced support of our proposed *PIA* methodology compared with other three *PIA* methodologies 117

6.1 The mapping between privacy design strategies and the NIST privacy controls 134
6.2 Function types and their weights 137
6.3 The privacy-enhancing profile with the «*enhance*» stereotype to express the privacy enhancement of a system model 139

7.1 *rabac* profile 158
7.2 An excerpt of the ClouDAT's control list 174

8.1 The privacy targets supported by the IDS and the privacy check (described in Chapter 4). 192
8.2 The table demonstrates to which threats of the *Treacherous 12*, the results of the three checks of CARiSMA refer. 193

B.1 The differences between the GDPR and Directive 95/46/EC 210

E.1 A mapping between privacy targets and the NIST privacy controls . 216

F.1 The privacy features of the *Minimize* strategy 218
F.2 The privacy features of the *Hide* strategy 219
F.3 The privacy features of the *Separate* strategy 220

F.4 The privacy features of the *Abstract* strategy 221
F.5 The privacy features of the *Inform* strategy 222
F.6 The privacy features of the *Control* strategy 223
F.7 The privacy features of the *Enforce* strategy 224
F.8 The privacy features of the *Demonstrate* strategy 224

Abbreviations

BSI	Bundesamt für Sicherheit in der Informationstechnik (German Federal Office for Information Security)
CNIL	Commission Nationale Informatique & Libertés (French Data Protection Authority)
CSA	Cloud Security Alliance
ENISA	European Union Agency for Network and Information Security
EU	European Union
GDPR	General Data Protection Regulation
IDS	Industrial Data Space
NIST	National Institute of Standards and Technology
PA	Public Administration
PbD	Privacy by Design
PET	Privacy Enhancing Technologies
PIA	Privacy Impact Assessment
PLA	Privacy Level Agreement
SSN	Social Security Number
UML	Unified Modeling Language
VDB	VisiOn Database
VPP	VisiOn Privacy Platform

Acknowledgements

During my PhD, I had the pleasure of working with outstanding people. In fact, undertaking this research has been a truly life-changing experience for me.

Firstly, I would like to express my sincere gratitude to my supervisor Prof. Dr. Jan Jürjens for all the support, encouragement, and motivation he gave me during this period. Jan guided me with planning the direction, finding my topic, and undertaking my research. I appreciate all his contributions of time, ideas, immense knowledge, and patience to make my PhD. I could not have imagined having a better supervisor for my research.

I wish to thank Prof. Dr. Patrick Delfmann for immediately agreeing to be my second referee, and his efforts and time to read my dissertation. My sincere thanks also go to the PhD committee.

I wish to thank Dr. Volker Riediger, and Dr. Daniel Strüber for our ongoing collaboration, their stimulating ideas, and their insightful suggestions. Volker's and Daniel's feedback and comments always allowed me to accomplish my tasks and provided me with invaluable perspectives on my work. Also, my deepest gratitude to Dr. Marco Konersmann, who kindly read my dissertation and made useful suggestions for its improvement.

I like to thank Katharina Großer, Jens Bürger, Sven Peldszus, and Qusai Ramadan who kindly proofread various chapters of this thesis and provided me with valuable Feedback.

I want to express my gratitude to all students, who with their promising works, provided me with valuable resources, including initial tool supports, system models, and case studies. I would like to thank the anonymous reviewers of all my publications. Their suggestions and comments were indispensable to improve this thesis. Furthermore, I wish to thank all my colleagues in two research projects ViSiOn and CLouDAT for our successful cooperations.

I would like to thank my parents, my two brothers (Arash and Ashkan) and all my friends, especially: Mehrdad, Ali, Shirin, Paria, and Mehdi who always supported me in my life.

Last but not least, I would like to thank Stephi who was always there for me in the last four years, and supported me with all her heart through every tough situation.

Finally, my sincere gratitude goes to all scientists, artists, and activists who strive to make our world a better place to live.

“Your life is your life.
Know it while you have it.”
Charles Bukowski

Chapter 1

Introduction

The number of IT systems that process personal data has increased dramatically in recent years [190]. Furthermore, concerning the significant technical advancement regarding data processing (such as big data analytics, online government services, and online social networks), the way in which data is processed no longer resembles the methods used around two decades ago [74]. For the data processors[1] that process the personal data of their service customers, a major challenge is to protect personal data. The consequences of failing to address this challenge are drastic and may result in significant damage to the data processors' reputation as well as finances, and cause personal and public embarrassment [158].

Many privacy risks arise when privacy concerns are not properly addressed during system development [54]. Imagine an administration office of a city that, to issue a birth certificate, requires the personal data of citizens. It stores the data in its database and in particular situations, it may transfer the personal data of citizens to other data processors[2]. The personal data of the citizens may be used to tailor advertisements to their interests. If it is stated that a citizen's personal data must not be processed for marketing purposes, a violation denoting *processing personal data for an unauthorized purpose* may arise. Such a violation leads to a privacy risk which jeopardizes the privacy of a citizen, as well as the reputation of a data processor. According to a special Eurobarometer report [74] on data protection, although 71% of the respondents agree that providing personal information is an increasing part of modern life, a majority of the respondents (53%) are uncomfortable about

[1] According to the General Data Protection Regulation [198] a processor is a legal person, public authority, agency, company, or other body which processes personal data.

[2] This example is based on one of the case studies of the VisiOn EU project—Visual Privacy Management in User-Centric Open Environment, `http://www.visioneuproject.eu/` (accessed: 2019-06-01). It is later used as a running example for this thesis.

using their personal information to tailor advertisements. The violation mentioned earlier (processing personal data for an unauthorized purpose) can be avoided by embedding privacy into the design of the IT system of the administration office at early phases of system development, for instance, by limiting the personal data processing only for specific processing purposes [58, 198].

The new data protection regulation of the EU—the General Data Protection Regulation (GDPR) [198]—requires to take privacy seriously from the onset of system development [88]. The prescription for *privacy by design* (Article 25 of the GDPR) is, in fact, one of the key changes in the GDPR. *Privacy by design* (*PbD*) obliges to integrate appropriate technical and organizational controls (such as pseudonymisation[3]) into the design from the early phases of the development in an effective manner to meet the requirements of the GDPR and protect the rights of data subjects. The GDPR came into effect in May 2018 repealing Directive 95/46/EC [197]. Infringements of the GDPR can be subject to administrative fines up to 4% of the total worldwide annual turnover (Article 83). Therefore, complying with the GDPR has become a top priority for organizations [19, 115].

With the applicability of the GDPR, *PbD* became an enforceable legal obligation. Since *PbD* requires to consider privacy concerns in the design of an IT system, it delegates the responsibility for privacy concerns to the developers of IT systems [98]. However, concerning the vague and ambiguous nature of regulations, the GDPR does not introduce a concrete and practical methodology or guidance to operationalize *PbD* [35, 96]. At first sight, the *PbD* principle (as per Article 25 of the GDPR) seems to simply prescribe integration of a few controls into an IT system during the development phase. However, to operationalize *PbD*, various aspects have to be considered. The difficulty of considering privacy concerns from the early stages of the system development emerges in the concept of privacy concerns itself, and how such concerns are defined [97]. Furthermore, the *European Data Protection Supervisor* (*EDPS*)—an independent institution of the EU—describes various dimensions of the obligation of *PbD* [76]: (I) *PbD* has to fully support the principles relating to the processing of personal data. (II) *PbD* has to consider privacy concerns from the early phases. (III) *PbD* has to identify concrete design violations, privacy threats and arising privacy risks. (IV) Besides risks, further factors such as the cost of implementation, the interrelations between controls and their effectiveness have to be considered when integrating the controls into a system. Hence, *PbD* is rather a powerful principle and includes more than the process of uptaking a few controls [188, 189].

In recent years, there is an increasing trend toward providing frameworks and approaches to consider privacy in the development process of an IT system. A num-

[3] To process personal data in a manner that personal data cannot be longer associated with a data subject without the use of additional data [142].

ber of approaches introduce privacy design strategies [52, 108], privacy design patterns [21, 52, 53, 80, 113, 160, 170, 183, 185] and privacy enhancing technologies [36, 58, 73, 83, 201], which provide strong privacy guarantees and assist system designers in protecting personal data. Such works are important to protect personal data, however, they do not rely on any analysis to explicitly identify where privacy is needed in a system. Moreover, they do not introduce a mechanism to identify privacy risks and choose appropriate strategies, patterns, or technologies to mitigate those risks.

Several commissioners and governments have proposed guidelines, practices, and recommendations on how privacy risks may be managed, and how *PbD* could be realized [59, 86, 122, 159]. One may benefit from the foundation of these works to facilitate the adoption of the *PbD* principle. However, they are rather abstract in nature and do not provide a practical methodology to identify concrete privacy violations and threats of a system.

There is a variety of approaches that provide model-based privacy-aware system development. In [65], the authors introduce LINDDUN, a methodology to identify certain privacy threats in data flow diagrams where such threats exploit privacy risks. In [23], an approach to build Unified Modeling Language (UML) models that specify and structure privacy concerns, thereby improving the privacy definition and enforcement is provided. PriS [131] represents a security requirements engineering method to incorporate privacy requirements early in the system development process. MAPaS introduces a promising model-based framework (based on UML) for the modeling and analysis of privacy-aware systems. It provides a set of analysis functions to assess domain models. In [13], an approach to express privacy related concepts in UML is provided. In [137], the authors propose an approach for model-driven privacy assessment in the Smart Grid. As stated in [54], several model-based approaches rely on *role-based access control* (*rbac*) models [181]. A classification of *rbac* approaches is presented in [172].

All these model-based approaches provide promising means to support privacy in the development process. However, the majority of them only focus on one of the necessary aspects (steps) mentioned above to operationalize *PbD*. They do not introduce a methodology to coherently support all aspects, namely defining privacy concerns, starting from the early phases, supporting the GDPR principles, identifying risks and choosing as well as integrating controls into systems. To operationalize *PbD*, a methodology is required, which adheres to the GDPR principles, particularly, the principles relating to the processing of personal data prescribed in Article 5. Moreover, since *PbD* aims at mitigating the privacy risks of a system by integrating appropriate controls from the early phases, identifying risks is a necessary step toward operationalizing *PbD*. This calls for a mechanism to identify privacy violations and threats which pose dangers to a system and cause privacy risks. Furthermore, as mentioned above, besides risks other issues such as the costs of risks mitigations have to be considered.

In this thesis, to address the lack of a rigorous methodology to operationalize *PbD*, covering the aspects discussed above, we propose a model-based methodology to operationalize *privacy by design*.

Model-based software engineering (MBSE) represents attractive mechanisms to systematically support the development process. MBSE has been established as a paradigm where models are the primary artifacts in the development of software systems. Developing complex systems is particularly challenging when different independent, or conflicting concerns such as privacy and security must be handled in those systems [85]. System models shield the developers from complexities through abstraction. A system model is an abstraction of some aspects of a system and allows a developer to focus on main concerns such as privacy and security [85]. Furthermore, model-based approaches cover the initial phases of development [54]. Various models, such as informal usage for communication or learning, semi-formal modeling for planning and documentation, and formal usage for generation, analysis and development, are widely used in industry and UML is the most used modeling language [193].

1.1 Challenges and Research Directions

Regarding our discussion above, we identify four challenges in this thesis:

- Initially, we need a basis to rely on when we take into account the processing of a piece of personal data. In other words, we need a means to define privacy concerns and to verify later whether they are supported when processing personal data.

- To integrate appropriate privacy controls into a system, we need to identify where precisely in a system such controls have to be applied.

- The privacy risks for the rights and freedoms of natural persons have to be determined. Identifying and mitigating privacy risks is at the core of the *PbD* principle. It has to be determined *what is at risk* in a system when processing personal data.

- Integrating appropriate privacy controls into a system in an effective manner is an intricate task. Apart from the privacy risks, the dependencies and interrelations between the controls as well as their costs have to be considered.

In the following sections, we introduce four main research directions of this thesis, which span over the above-mentioned challenges and formulate the research questions (*RQ*s).

1.1.1 Privacy Preferences

To consider privacy concerns from the early phases of system development, first, it has to be specified what are the privacy concerns. When it is claimed that privacy has to be protected, it is unclear what is precisely meant [186]. Generally, privacy is a difficult notion to define [146]. As a legal concept, there is no specific definition of privacy [97]. Article 5 of the GDPR prescribes a set of principles relating to the processing of personal data. However, this does not particularly specify a means to define privacy concerns. In this thesis, we use the term *privacy preference*s when we talk of privacy concerns. A formal definition to specify privacy preferences is required. This leads to our first research question:

RQ1: *How can privacy preferences be defined?*

Moreover, concerning the fact that a piece of personal data may be processed by several data processors, to systematically denote the privacy preferences, agreements on the use of personal data between data processors are required. Besides privacy preferences, the agreements include privacy violations, threats, risks and controls to mitigate the identified risks.

RQ2: *How can agreements on the use of personal data be established to systematically specify the privacy preferences and support a privacy analysis?*

1.1.2 Privacy Analysis

The second challenge calls for system analysis. Concerning the example provided at the beginning of this chapter, a piece of personal data may be processed by several data processors. Thus, an analysis may require to analyze several systems. This leads to the following research question:

RQ3: *How can an analysis be performed on a system in an environment where a piece of personal data is processed by several data processors?*

A system analysis has to verify whether privacy preferences are properly supported when processing personal data. The results of an analysis denote the privacy violations of a system. Without knowing these violations, one cannot effectively integrate appropriate controls into a system (from early phases of the system development) to protect the privacy of personal data. Performing analysis of a system requires a specification of the system. In this thesis, the systems are specified by system models. As mentioned previously, system models address the complexity of the systems by abstraction and enable the analysis of the systems from early phases. UML [157] is used to model a system. The term system model in this thesis refers to a set of UML diagrams that model the structure and behavior of a system. A methodology to analyze system models regarding privacy preferences is required. This leads to the research question **RQ4**. The privacy preferences in this thesis are defined based on four key elements of privacy, namely purpose, visibility, granularity, and retention (introduced in Barker et al.'s seminal taxonomy [22]). Therefore, in **RQ4**, we investigate those key elements of privacy.

RQ4: *How can a system design that processes personal data be analyzed to verify whether the key elements of privacy are supported?*

1.1.3 Privacy Impact Assessment

The controls—to be integrated into the system design to fulfill *PbD*—shall be identified, taking into account the privacy risks [76, 198]. Hence, a risk assessment approach (*privacy impact assessment* (*PIA*) as per Article 35 of the GDPR) with a view to selecting and implementing controls for effective protection is necessary. To identify risks, one important step is to identify threats. A threat is a potential cause of a violation and may pose a privacy risk.

Concerning these, we investigate the following research questions:

RQ5: *Given a system model, how can concrete privacy threats be identified?*

RQ6: *How can a privacy impact assessment be conducted to identify the privacy risks?*

1.1.4 Privacy Enhancement

Integrating an appropriate set of controls into a system design involves a number of sensitive aspects [76]. Besides privacy risks, the interrelations between the controls as well as the costs of the controls have to be taken into account. Moreover, the controls have to be incorporated into to system models. This enables one to iteratively analyze the systems, thereby verifying whether the risks are handled properly.

This gives rise to two research questions:

RQ7: *How can an adequate selection of controls (concerning varying risks, interrelations between controls and the costs of controls) be identified to mitigate the identified privacy risks?*

RQ8: *How can the selected controls be incorporated into a system model?*

1.2 Contributions

Figure 1.1 sketches the foundation of our methodology. The focus of this figure is only to illustrate our contributions; we do not show further input or output artifacts. Below, eight items shortly describe our contributions in this thesis. The items address the eight research questions, respectively.

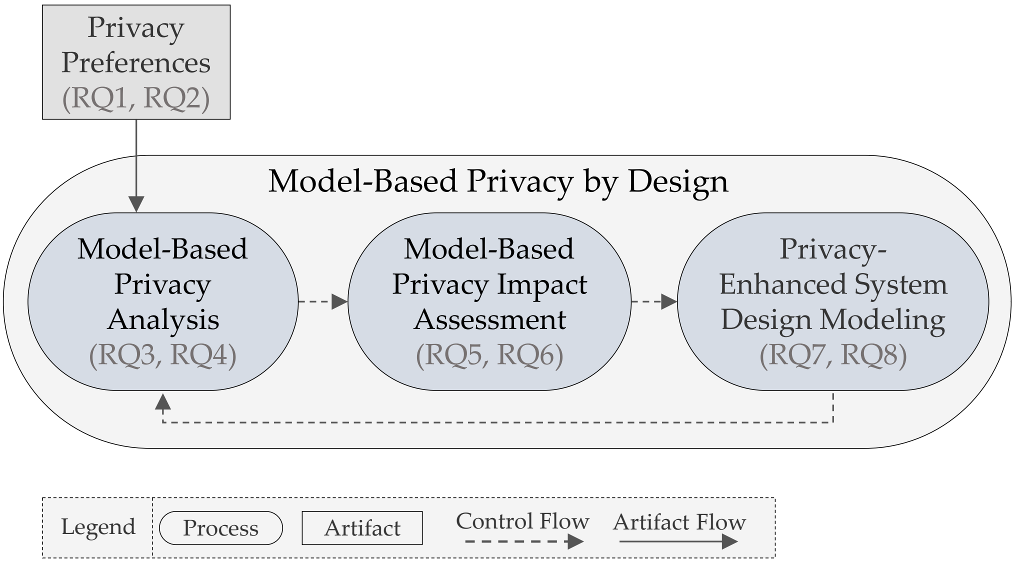

Privacy Preferences (RQ1, RQ2)
Model-Based Privacy by Design
Model-Based Privacy Analysis (RQ3, RQ4)
Model-Based Privacy Impact Assessment (RQ5, RQ6)
Privacy-Enhanced System Design Modeling (RQ7, RQ8)
Legend
Process
Artifact
Control Flow
Artifact Flow

Figure 1.1: The foundation of our *PbD* methodology focusing only on our contributions in this thesis. The figure shows how our contributions address the research questions introduced in Section 1.1.

Privacy Preferences (Chapter 3):

- We provide a definition as well as a foundation for privacy preferences. The privacy preferences are defined based on four key elements of privacy, namely purpose, visibility, granularity, and retention (**RQ1**).
- We leverage the *privacy level agreement* (*PLA*) outline, which is originally introduced by the *Cloud Security Alliance*[4], and extend it to support the GDPR fully. We provide a metamodel to specify the structure of PLAs (**RQ2**).

Model-Based Privacy Analysis (Chapter 4):

- We introduce a modular analysis methodology that separately analyzes the system design of the data processors which cooperatively process a piece of personal data (**RQ3**).
- We provide a model-based methodology to analyze the design of an IT system in regard to a set of privacy preferences. Our analysis relies on several privacy checks to verify a system model. To enable such an analysis, a mechanism to express privacy concerns in the system models is introduced (**RQ4**).

Model-Based Privacy Impact Assessment (Chapter 5):

- To identify the concrete privacy threats in a system model, the results of a model-based privacy analysis, which denote a set of privacy design violations, are further evaluated (**RQ5**).
- We explain how our proposed model-based privacy analysis methodology supports a *privacy impact assessment*. We consider a set of privacy targets (privacy targets are derived from the privacy principles introduced by the GDPR) and investigate how the results of a model-based privacy analysis pose risks to the privacy targets (**RQ6**).

Privacy-Enhanced System Design Modeling (Chapter 6):

- We propose a systematic model-based methodology to coherently perform the privacy enhancement of IT systems taking into account the privacy risks, the interrelations between the controls and their costs (**RQ7**).

[4] `https://cloudsecurityalliance.org/` (accessed: 2019-06-01)

- We show how controls are integrated into system models in different abstraction levels. We use a mechanism to capture the extensive variety of privacy controls. We further use and extend a model-based approach to estimate the costs of the controls (**RQ8**).

1.3 Methodology

We use the *design science* research to conduct our work in this thesis. Design science research is a constructive research paradigm which seeks to extend the boundaries of human and organizational capabilities by creating new and innovative artifacts such as models, methods, theories, instantiations, algorithms and system design methodologies [57, 106, 107]. Constructive research offers both practical and theoretical results, and addresses different problems, regarding *novelty*, *feasibility*, and *improvement*.

Hevner et al. [107] developed a conceptual framework, including seven guidelines, for conducting and evaluating good design science research:

> **Guideline 1**: Design science research must produce viable artifacts such as a model or a method (*design as an artifact*).
>
> **Guideline 2**: A relevant problem has to be solved (*problem relevance*).
>
> **Guideline 3**: The utility, quality, and efficacy of the designed artifacts have to be demonstrated via well-executed evaluation methods (*design evaluation*).
>
> **Guideline 4**: Clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies have to be provided (*research contributions*).
>
> **Guideline 5**: Design science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact. (*research rigor*).
>
> **Guideline 6**: The search for an effective artifact requires utilizing available means to reach desired ends (*design as a search process*).
>
> **Guideline 7**: The design science research must be presented effectively (*communication of research*).

In this thesis, we provide a model-based methodology to operationalize *privacy by design*. The methodology itself comprises different sub-methodologies, models, and artifacts (Guideline 1). Earlier in this chapter, the importance and the challenges of *PbD* are demonstrated (Guideline 2). We use different techniques to evaluate our work. Below, we discuss the evaluation techniques in this thesis (Guideline 3). The clear contributions of this thesis are demonstrated in Figure 1.1 (Guideline 4). Our concepts proposed in this thesis benefit from various existing works such as model-based security analysis by Jürjens [128], the theory of sets [123] and lattices [150], a data privacy taxonomy [22], BSI (German Federal Office for Information Security) privacy impact assessment guideline [159] and feature modeling [133] (Guideline 5). Design is a search process to identify a proper solution to a problem [107]. In our research, we first identified four challenges and to achieve our main aim (realizing *PbD*) we introduce three sub-methodologies. We demonstrate the capabilities of our concepts with various case studies and use several evaluation techniques. Finally, based on limitations, we argue the future research directions (Guideline 6). Finally, this thesis provides structured documentation of our work (Guideline 7).

The research methodology conducting this thesis is explained in the taxonomy of software engineering proposed by Shawn [184]. In this taxonomy, concerning the results of a research project, Shawn distinguishes five approaches to address a software engineering problem, namely *qualitative or descriptive model*, *technique*, *system*, *empirical predictive model* and *analytic model*.

In this thesis, different *techniques*, supported by *analytic models*, address the research challenges.

> "**Technique**: Invent new ways to do some tasks, including procedures and implementation techniques [...]"
> "**Analytic model**: Develop structural (quantitative or symbolic) models that permit formal analysis."

We provide novel automated techniques to conduct a privacy analysis and a privacy impact assessment, and to enhance a system design with privacy controls. In Chapter 4, the privacy analysis uses a modular method to analyze the system models and is enabled by extended privacy level agreements. Furthermore, in Chapter 5, an extended list of privacy targets and a novel method to calculate severities, facilitate our proposed privacy impact assessment methodology. In Chapter 6, we leverage a new model-based cost estimation approach, a feature model and extended aspect models to enhance a system design with privacy controls.

Five techniques **to validate the software engineering results** are explained in [184], namely *persuasion*, *analysis*, *implementation*, *evaluation*, and *experience*. In this thesis, we apply all these techniques.

Persuasion is used to motivate the methodologies and design choices throughout this thesis. In Chapters 3 and 4, we use *analytic proofs* to argue for correctness (regarding our definition of the privacy preferences and our proposed privacy checks). In Chapter 7, we discuss the tool support (*implementation*) for our concepts presented in this thesis. In Chapter 4, based on the results of a survey and our observations (*experience*), we discuss and investigate the support required by the users of our proposed model-based privacy analysis. In Chapter 5, *comparative evaluation* is used to compare our proposed privacy impact assessment methodology with the existing legal methodologies. In Chapters 4, 5, 6, and 8, we use practical case studies to *evaluate* our concepts, thereby showing their applicability in real-world settings.

1.4 Thesis Outline

The remainder of this thesis is structured as follows:

- In Chapter 2, we describe the overall workflow of this thesis. We further introduce a running example.
- In Chapter 3, we define the privacy preferences and extend *privacy level agreement*s introduced by the Cloud Security Alliance (*CSA*) to manifest the privacy preferences. We further specify the structure of the agreements by providing a metamodel.
- In Chapter 4, we propose a modular model-based privacy analysis methodology. We further investigate the support required by the users of this methodology.
- In Chapter 5, we propose a methodology to support a *privacy impact assessment* (*PIA*) by performing model-based privacy analysis.
- In Chapter 6, we propose a methodology to support the coherent privacy enhancement of a system design model. The enhancement is performed concerning an extensive variety of privacy controls, including privacy-design strategies, patterns and privacy enhancing technologies.
- In Chapter 7, we discuss the tool support for our proposed concepts in this thesis. This section includes our contributions to two research projects: namely VisiOn and ClouDAT.
- In Chapter 8, to further evaluate the applicability of our proposed model-based privacy analysis methodology in an environment comprising several

data processors, we apply it to the industrial data space (IDS). The IDS provides a basis for creating and using smart services while ensuring digital sovereignty of service customers.

- In Chapter 9, we conclude. Moreover, we discuss the assumptions and limitations of our model-based *PbD* methodology. Finally, we outline possible future research directions.

1.5 How to Read this PhD Thesis

In Figure 1.2, we visualize the outline of this thesis and show where (by chapter) our research questions are addressed. The four conceptual Chapters 3, 4, 5 and 6 that answers the research questions are surrounded in a box.

This thesis is structured in a way that each of these four chapters may be read separately. We provide a separate motivation, introduction, related work, validation, discussion and conclusion for each. However, the thesis benefits from a running example and a storyline (Chapter 2) and to perceive the main contribution (a *model-based privacy by design*) one has to follow the four chapters sequentially.

One has to:

- First, understand how privacy preferences are defined (Chapter 3).
- Then, continue by understanding the process of performing a model-based privacy analysis concerning privacy preferences (Chapter 4).
- Afterwards, relying on the results of a privacy analysis (the identified privacy design violations), get to know the process of conducting a *privacy impact assessment* to identify the privacy risks (Chapter 5).
- And finally, with respect to the last three steps, understand how a system may be enhanced by a set of privacy controls in the early phases of design, which is, in fact, the main aim of the *PbD* principle (Chapter 6).

As mentioned above, each of the conceptual chapters is validated separately using the techniques described in Section 1.3. In Chapter 8, we additionally apply our model-based privacy analysis to the industrial data space (IDS).

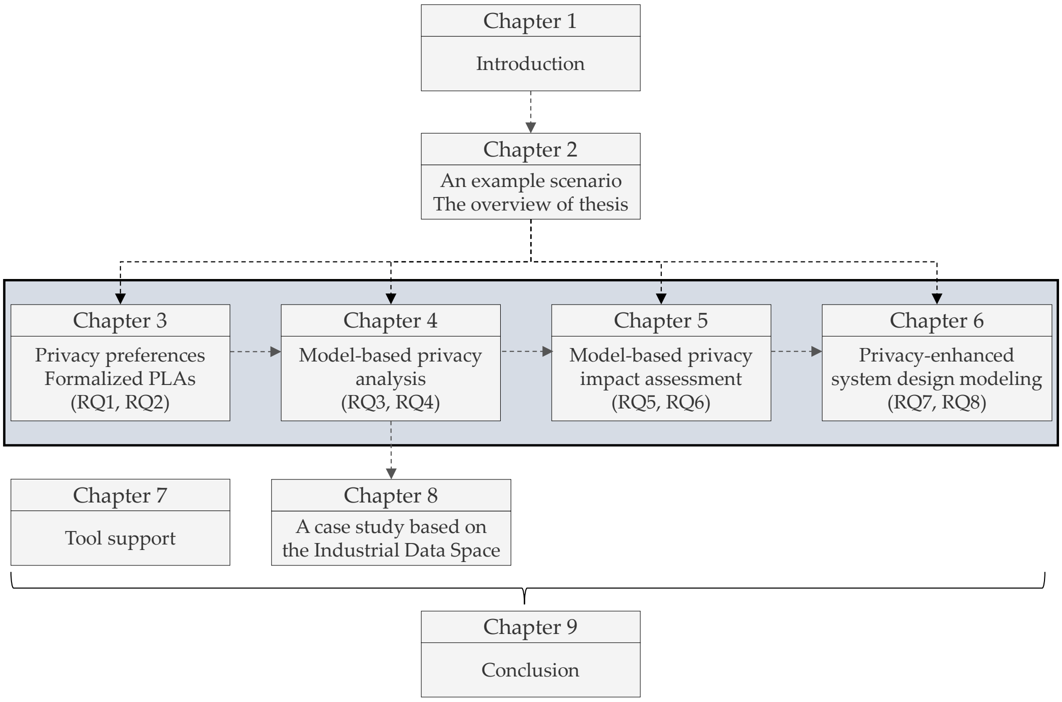


Figure 1.2: The outline of the thesis, showing how to read this thesis

1.6 Preliminary Publications

This thesis shares material with eight research papers written by the author of this thesis. The co-authors have explicitly confirmed the individual contributions of the author of this thesis to these papers[5].

- Amir Shayan Ahmadian, Daniel Strüber, and Jan Jürjens. Privacy-enhanced system design modeling based on privacy features. In Proceedings of the 34th Annual ACM Symposium on Applied Computing, SAC 2019, Limassol, Cyprus, April 08-12, 2019, pages 1492–1499, 2019.
- Amir Shayan Ahmadian, Daniel Strüber, Jan Jürjens, and Volker Riediger. Model-based privacy analysis in industrial ecosystems: A formal foundation. International Journal on Software and Systems Modeling. Submitted.
- Amir Shayan Ahmadian, Daniel Strüber, Volker Riediger, and Jan Jürjens. Supporting privacy impact assessment by model-based privacy analysis. In Proceedings of the 33rd Annual ACM Symposium on Applied Computing, SAC 2018, Pau, France, April 09-13, 2018, pages 1467–1474, 2018.

[5] The signed confirmations are submitted to the PhD committee.

- Amir Shayan Ahmadian, Jan Jürjens, and Daniel Strüber. Extending model based privacy analysis for the industrial data space by exploiting privacy level agreements. In Proceedings of the 33rd Annual ACM Symposium on Applied Computing, SAC 2018, Pau, France, April 09-13, 2018, pages 1142–1149, 2018.

- Amir Shayan Ahmadian, Sven Peldszus, Qusai Ramadan, and Jan Jürjens. Model-based privacy and security analysis with CARiSMA. In Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering, ES-EC/FSE 2017, Paderborn, Germany, September 4-8, 2017, pages 989–993, 2017.

- Amir Shayan Ahmadian, Daniel Strüber, Volker Riediger, and Jan Jürjens. Model-based privacy analysis in industrial ecosystems. In Modelling Foundations and Applications - 13th European Conference, ECMFA 2017, Held as Part of STAF 2017, Marburg, Germany, July 19-20, 2017, Proceedings, pages 215–231, 2017.

- Amir Shayan Ahmadian and Jan Jürjens. Supporting model-based privacy analysis by exploiting privacy level agreements. In 2016 IEEE International Conference on Cloud Computing Technology and Science, CloudCom 2016, Luxembourg, December 12-15, 2016, pages 360–365, 2016.

- Amir Shayan Ahmadian, Fabian Coerschulte, and Jan Jürjens. Supporting the security certification and privacy level agreements in the context of clouds. In Business Modeling and Software Design - 5th International Symposium, BMSD 2015, Milan, Italy, July 6-8, 2015, Revised Selected Papers, pages 80–95, 2015.

Chapter 2

Model-Based Privacy by Design: An Overview of the Methodology

In this chapter, we first introduce a terminology to explain the key terms that are used in this thesis. Afterwards, we introduce a scenario based on a practical case study which is used as a running example in this thesis. We phrase our research questions, introduced in the previous chapter, concerning this example. Finally, to overview what this thesis is about, we describe the overall workflow of the methodology proposed in this thesis.

2.1 The Common Terms in this Thesis

In this section, we overview the terms that are heavily mentioned in the rest of this thesis. Appendix A, besides these terms, lists the definitions and terms that we later introduce throughout this thesis.

Personal data (the GDPR, Article 4, paragraph 1) means any information relating to an identified or identifiable natural person. In Chapters 4 and 5, we use the term *sensitive data*. Sensitive data particularly adheres to the definition of several categories of personal data including the above-mentioned definition, *special categories* of personal data (the GDPR, Article 9), *general identification number* (the GDPR, Article 87) and *privacy-relevant data* [58].

With the term *processing*, similar to the GDPR (Article 4, paragraph 2), we mean any operation performed on personal data such as collection, recording, organization, structuring, storage, adaptation or alteration, retrieval, consultation, use, disclosure by transmission, dissemination or otherwise making available, alignment or combination, restriction, erasure or destruction.

According to the GDPR, a *data controller* determines the purposes and the means for the processing of personal data. In our work, a *service customer* is a data controller, who provides personal data. A *data processor* processes personal data on behalf of the controller. When we talk of *service providers*, we mean either a data processor who directly processes the provided data, or a data controller who transfers the data to other data processors. In this thesis, both the service customers and service providers are *organizations* (not a natural person).

In the example scenario that we introduce in the following section, a group of organizations (several data controllers and data processors such as enterprises, public administrations and financial institutes) are engaged in the processing of personal data. In the rest of this thesis, we call such a group of organizations an *industrial ecosystem*.

The terms *privacy* and *data protection* refer to different meanings in the EU legal framework. In the Charter of Fundamental Rights of the EU [78], *privacy* is used to describe Article 7:

> "Everyone has the right to respect for his or her private and family life [...]."

whereas *data protection* is stipulated in Article 8:

> "Everyone has the right to the protection of personal data concerning him or her."

With respect to the preliminary opinion of the *European Data Protection Supervisor* (*EDPS*) on *privacy by design* [76] and Article 25 of the GDPR—on data protection by design and by default—in this thesis, when we refer to *privacy by design*, we also comprise any uses of *data protection by design*. Moreover, *privacy by design* does not exclude *privacy by default*, but just gives special importance to the *design* phase [76].

2.2 Running Example

Consider the process of *issuing a birth certificate* in an administration office. This process belongs to a practical case study, namely the *birth certificate registration* scenario in *Municipality of Athens* (*MoA*). MoA is a *public administration* (*PA*) in the city

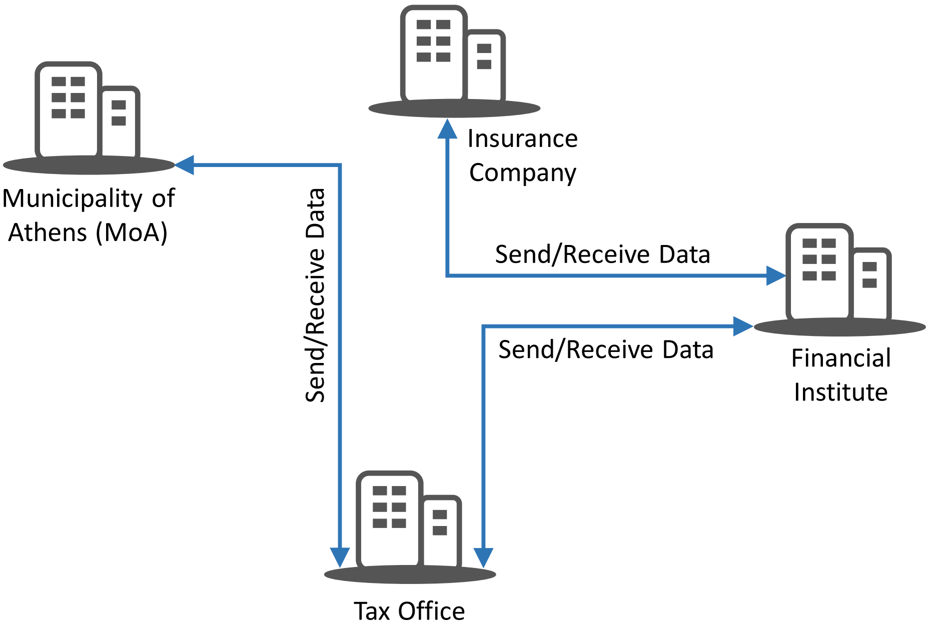


Figure 2.1: An illustration of the industrial ecosystem where MoA (*Municipality of Athens*) sends personal data to other service providers

of Athens. This case study is one of the case studies of the *VisiOn*[1,2,3] [68] research project.

In the VisiOn (Visual Privacy Management in User-Centric Open Environment) project, a platform (the VPP, Visual Privacy Platform) is developed to assist public administrations, such as the administration office of a city or a hospital, to design IT systems that take privacy concerns into account. The VPP further guides citizens to specify their privacy concerns and to control how their personal data is processed.

DAEM[4], the IT company of *Municipality of Athens* is in the process of developing an online service to issue a birth certificate. The *issuing a birth certificate* process requires the *Social Security Number* (*SSN*) of a citizen to perform the processing. The *SSN* (*AMKA* [1] in Greek) is the insurance ID of a person in Greece. Following the definition of personal data in Section 2.1, the *SSN* is a piece of personal data. To operationalize *PbD*, DAEM needs to consider a set of privacy concerns when designing the *issuing a birth certificate* process and when necessary have to integrate

[1] `http://www.visioneuproject.eu/` (accessed: 2019-06-01)

[2] `http://ec.europa.eu/research/infocentre/article_en.cfm?artid=46216` (accessed: 2019-06-01)

[3] `https://cordis.europa.eu/project/rcn/194888_en.html` (accessed: 2019-06-01)

[4] `http://www.daem.gr/` (accessed: 2019-06-01)

appropriate controls into the system design.

The processing is mainly performed within the MoA's system, however, the *SSN* may be transferred to other service providers for further processing. Figure 2.1 illustrates the industrial ecosystem, where MoA transfers data (including the *SSN*) to other service providers. MoA sends the citizen's *SSN* to a tax office to verify the tax status of the citizen. The tax office may additionally need to verify the solvency of the citizen. Therefore, it transfers the *SSN* to a financial institute. The financial institute may further need to query an insurance company. Although from the perspective of a citizen only MoA processes the *SSN*, the *SSN* is, in fact, processed by several service providers.

We introduced a set of research questions in Section 1.1. We phrase our research questions concerning the *issuing a birth certificate* scenario. As previously mentioned, in this thesis, a system is modeled by various UML diagrams specifying the structure and behavior of the system.

- *How can privacy preferences be defined for the SSN? (RQ1)*

  In the first step, a foundation (a set of conditions) has to be identified which enables one (DAEM) to verify whether the processing of the *SSN* is authorized. Article 5 of the GDPR prescribes a set of principles relating to the processing of personal data. These principles provide a foundation to specify a set of privacy preferences relating to the processing of the *SSN*.

- *How can agreements on the use of personal data (including the SSN) be established to specify the privacy preferences and support a privacy analysis of the MoA and the tax office systems? (RQ2)*

  MoA may send the personal data (such as the *SSN*) to several other data processors. A means (such as an agreement) to capture and formally specify how the *SSN* may be processed is required. MoA not only processes the *SSN*, but it deals with various kinds of personal data. Therefore, such agreements have to consider the whole personal data which are processed by MoA, the tax office, and other service providers. These agreements enable one to conduct a privacy analysis regarding the specified conditions on the use of personal data and ensure the legitimacy of personal data processing.

- *How can an analysis be performed in the environment demonstrated in Figure 2.1, where to issue a birth certificate, different service providers need to process the SSN? (RQ3)*

The *SSN* is processed by several service providers. Since the privacy preferences on the processing of the *SSN* may not be identical in each service provider and the system models of not all service providers may be available to conduct a privacy analysis, each service provider has to be analyzed individually—in a modular method.

- *How can the systems' design of service providers that process the SSN be analyzed to verify whether the key elements of privacy are supported? (RQ4)*

A system model analysis is required to verify whether the processing of *SSN* conforms to the associated privacy preferences (defined based on the key elements of privacy). Such an analysis identifies the potential privacy design violations in a system.

- *Concerning the system model of MoA, how can privacy threats be identified? (RQ5)*

In the process of *issuing a birth certificate*, it has to be verified which privacy threats may arise due to the (potential) unauthorized processing of the *SSN*.

- *How can a privacy impact assessment be conducted to identify the privacy risks of MoA? (RQ6)*

Concerning the identified privacy design violations, a *privacy impact assessment* has to be conducted to identify *"what is at risk"* when processing the *SSN*. The results of this step are a set of privacy risks denoting how a system may be endangered when processing personal data.

- *How can an adequate selection of controls (concerning varying risks, interrelations between controls, and the costs of controls) be identified to mitigate the privacy risks of processing the SSN in the issuing a birth certificate scenario? (RQ7)*

Furthermore, after performing a *privacy impact assessment* and to mitigate the privacy risks, a set of appropriate controls has to be identified. Choosing appropriate controls to mitigate the risks is, however, a complex task which includes a set of aspects such as risks, varying costs, and the interrelations between the controls.

- *How can the selected controls be incorporated into the system model of MoA? (RQ8)*

> Eventually, the identified controls have to be integrated into the system design of the service providers. Therefore, a methodology is required to enhance the system models of the analyzed service providers in a way that privacy preferences are supported.

These questions represented the difficulties that DAEM faces when it aims to accomplish the core of *PbD*, which is the integration of appropriate controls from the early stages of system development. In this thesis, we introduce a model-based privacy by design methodology to operationalize *PbD*, thereby answering these questions.

2.3 Walk-Through: Model-Based Privacy by Design

Figure 2.2 presents the overall workflow of this thesis. The methodology that is described in this work takes multiple input artifacts, performs various analyses and enhancements, thereby generating several output artifacts. This figure only provides an abstract overview of this thesis.

PbD is operationalized by three sub-methodologies namely *model-based privacy analysis*, *model-based privacy impact assessment*, and *privacy-enhanced system design modeling*. According to Figure 2.2, these sub-methodologies are demonstrated as three sequential processes that have to be performed to accomplish the core of *PbD*—integrating appropriate controls into a system. *UML* (*Unified Modeling Language*) [157] is used to model the systems. The whole methodology is enabled by the definition of privacy preferences. In the following sections, we elaborate on various parts of Figure 2.2.

2.3.1 Privacy Preferences (Chapter 3)

As motivated in Section 1.1, the first step to operationalize *PbD* is to identify what privacy concerns have to be taken into account. Following Section 2.1, a data controller specifies the purposes and the means of processing personal data which have to be protected when processing personal data. The purposes and the means of processing are, in fact, the privacy concerns that have to be considered from the onset of system development. We use the term privacy preferences to specify the purposes and the means of processing personal data. A set of principles related

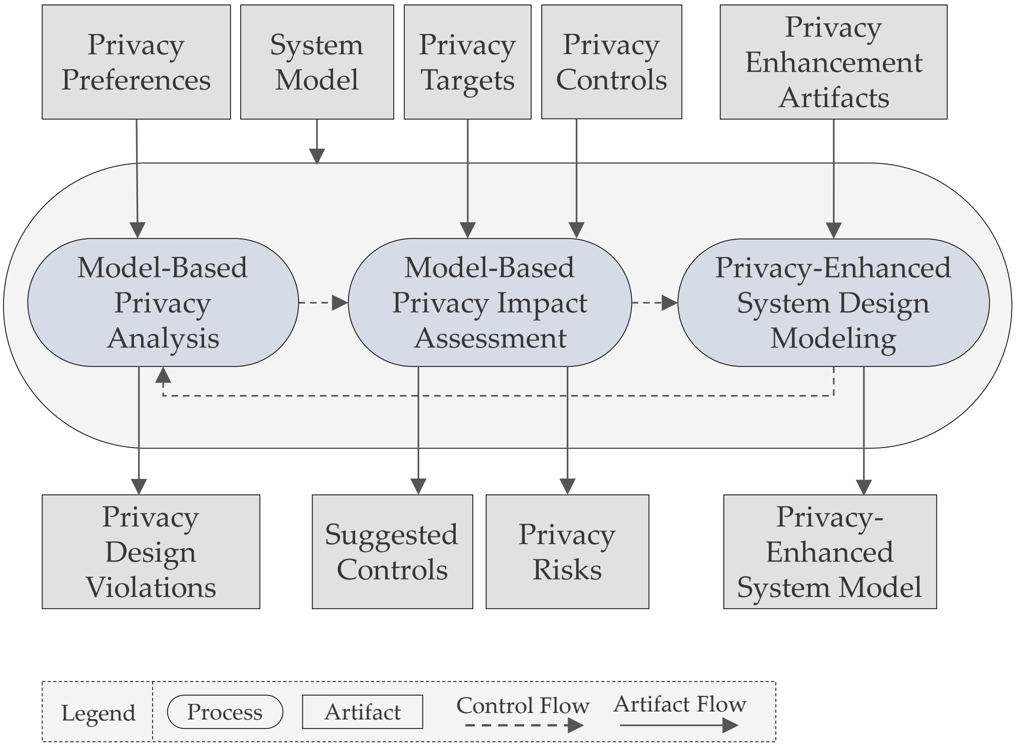

Privacy Preferences
System Model
Privacy Targets
Privacy Controls
Privacy Enhancement Artifacts
Model-Based Privacy Analysis
Model-Based Privacy Impact Assessment
Privacy-Enhanced System Design Modeling
Privacy Design Violations
Suggested Controls
Privacy Risks
Privacy-Enhanced System Model
Legend
Process
Artifact
Control Flow
Artifact Flow

Figure 2.2: The overall workflow of the model-based privacy by design methodology

to the processing of personal data is introduced in Article 5 of the GDPR. Pursuant to these principles and concerning the four fundamental privacy elements (introduced in [22]), namely *purpose*, *visibility*, *granularity*, and *retention*, we define privacy preferences in Chapter 3.

Furthermore, we motivated the need for agreements on the use of personal data which specify how personal data are authorized to be processed in industrial ecosystems. We benefit from the *privacy level agreements* (*PLA*s) outline [49] introduced by the Cloud Security Alliance to establish such agreements. Since PLAs are originally based on the former privacy regulation of the EU (*Directive 95/46/EC*), we update the PLA outline to support the GDPR fully.

The privacy preferences have to be specified before performing a privacy analysis. A PLA is concluded between a data controller and a data processor. It includes the privacy preferences of personal data.

2.3.2 Model-Based Privacy Analysis (Chapter 4)

A system model analysis in regard to the privacy preferences has to be performed to verify whether privacy preferences are met when processing a piece of personal data. Such an analysis denotes the potential violations in a system model. According to Figure 2.2, a privacy analysis takes as input a system model and a set of privacy preferences.

To conduct an analysis, the privacy elements have to be expressed in the system model. Since we use UML to model a system, we introduce a privacy profile to annotate the UML diagrams with the privacy elements. The annotation of a system model is performed manually by a system designer before performing an analysis, however, in Section 7.2.2.2, tool support to assist a designer in annotating a model is provided.

2.3.3 Model-Based Privacy Impact Assessment (Chapter 5)

As previously motivated, one key aspect of achieving *PbD* is taking into account the risks for service customers caused by personal data processing. The GDPR in Article 35 prescribes *Privacy Impact assessment* (*PIA*), which requires an assessment of the impact of personal data processing on the privacy of personal data and mitigates the arising privacy risks by suggesting appropriate controls.

Despite the existence of several national legal methodologies such as BSI (German Federal Office for Information Security) *PIA* methodology [159], CNIL (French Data Protection Authority) *PIA* methodology [86] and the UK *PIA* code of practice [59], the *PIA* adoption is still rare [158]. Moreover, there is no methodology to consider the concrete design of a system to identify specific privacy design violations, harmful activities, and threats. The legal methodologies describe a set of generic and abstract steps toward *PIA*s and are not suitable to be a process reference model.

In this thesis, we introduce a *PIA* methodology which benefits from our proposed model-based privacy analysis (Section 2.3.2). Our *PIA* methodology is based on the BSI *PIA* methodology [159] and takes as input the privacy design violations (the results of the privacy analysis), a set of privacy targets and a set of privacy controls (see Figure 2.2). Privacy targets are derived from the privacy principles and provide concrete and auditable bases to perform a risk assessment [158, 159]. Our *PIA* methodology enables a system designer to identify the privacy risks after performing a model-based privacy analysis. Following performing a *PIA*, the privacy targets that are endangered are identified. Upon this, the *PIA* methodology suggests a set of controls to mitigate those risks.

We map the violations resulting from performing a privacy analysis to a set of threats. To fully support the GDPR, we extend the list of privacy targets proposed by BSI with new privacy targets. Moreover, we describe how the results of a privacy analysis may be associated with the extended list of privacy targets to discover *"what is at risk."* Considering the privacy targets that are at risk, we introduce a method to assess the risks, based on the feedback from the data controllers and data processors (the public embarrassment of service customers and the reputation of the service providers). Eventually, with respect to the identified risks, a set of controls are suggested to mitigate the risks of processing personal data. Privacy level agreements are used to document *PIA* reports.

2.3.4 Privacy-Enhanced System Design Modeling (Chapter 6)

After performing a *PIA*, an appropriate selection of privacy controls have to be incorporated into a system model. As demonstrated in Figure 2.2, we provide a methodology to enhance a system model following conducting a *PIA*. Our privacy enhancement particularly supports three crucial types of inputs:

- Risks to the rights of natural persons by considering the results of the *PIA* in privacy enhancement.
- The interrelation and dependencies among the privacy controls.
- The costs of the controls.

Since privacy controls (such as *NIST* privacy controls [154] informed by *National Institute of Standards and Technology*[5]) are too generic to be directly integrated into a system model; we map them to a set of functionally enforceable strategies, patterns, and technologies. We use feature modeling [133] to capture an extensive variety of strategies, patterns, and technologies; and specify the interrelations between them. Moreover, to estimate the varying costs of incorporating controls into a system model, we propose a model-based methodology to estimate the costs of privacy enhancement. This methodology relies on counting certain elements in UML diagrams (particularly, activity diagrams). The feature model and the cost model are demonstrated as *privacy enhancement artifacts* in Figure 2.2.

Eventually, to integrate strategies, patterns, and technologies into a system model, we extend our proposed UML profile (Section 2.3.2) to establish traceability between model elements and the privacy controls. We further use and extend the

[5] `https://www.nist.gov/` (accessed: 2019-06-01)

concepts of *Reusable Aspect Modeling* [136] to express privacy enhancement in a system model. The result of our methodology is a privacy-enhanced system model.

As demonstrated in Figure 2.2, the process of identifying violations, assessing the risks and enhancing a system model may be iteratively continued to verify whether existing privacy design violations and arising risks are properly mitigated through applying the controls. Concerning this, in fact, our model-based privacy by design methodology may be applied to existing systems as well. In this case, the practitioners may analyze existing systems to identify potential violations and risks, and enhance the level of privacy protection.

Chapter 3

Privacy Preferences: A Foundation

This chapter shares material with the CloudCom'16 paper "Supporting Model-Based Privacy Analysis by Exploiting Privacy Level Agreements" [3] and the paper "Model-Based Privacy Analysis in Industrial Ecosystems: A Formal Foundation" [4] submitted to the SoSym Journal.

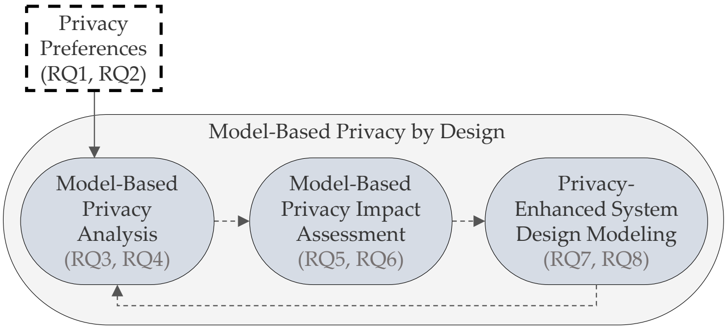

Privacy Preferences (RQ1, RQ2)
Model-Based Privacy by Design
Model-Based Privacy Analysis (RQ3, RQ4)
Model-Based Privacy Impact Assessment (RQ5, RQ6)
Privacy-Enhanced System Design Modeling (RQ7, RQ8)

Figure 3.1: The highlighted section (dashed lines) denotes how Chapter 3 contributes to the overall workflow (introduced in Section 2.3).

In Chapter 1, we mentioned that the first step toward operationalizing *privacy by design* (*PbD*) is to specify privacy concerns. We stated that we use the term *privacy preferences* when we talk of privacy concerns. In this chapter, we explain how the privacy preferences of a piece of personal data are specified. Privacy preferences provide a basis to conduct a privacy analysis. Our definition of privacy preferences is based on the four key privacy elements, namely *purpose*, *visibility*, *granularity*, and *retention* [22]. We present a foundation for our proposed privacy preferences based on the set theory. Moreover, we use the *privacy level agreement* (*PLA*) outline [49] (originally introduced by the *Cloud Security Alliance*) to capture the *privacy prefer-*

*ence*s of the *data controller*s. We propose a metamodel to formalize the structure of the PLA outline.

3.1 Introduction

The service customers (data controllers) determine the privacy preferences. The service providers (data processors, or data controllers) have to ensure that the privacy preferences are supported by their systems [198]. To verify this, the IT systems of the service providers have to be analyzed in regard to the privacy preferences [7]. To perform such an analysis, the privacy preferences have to be defined beforehand.

To systematically denote the privacy preferences of personal data used by a data processor, a mechanism to formally capture the privacy preferences is required. This could be achieved by specifying agreements on the use of personal data between a data controller and a data processor.

Motivated by this, we investigate the following research questions in this chapter:

RQ1: *How can privacy preferences be defined?*

RQ2: *How can agreements on the use of personal data be established to systematically specify the privacy preferences and support a privacy analysis?*

To address these research questions, we make the following contributions:

- Based on the four key privacy elements, namely *purpose*, *visibility*, *granularity*, and *retention*, we define privacy preferences (Section 3.3).
- We use the *privacy level agreement* (*PLA*) outline [49], introduced by the Cloud Security Alliance, to conclude agreements on the use of personal data between data controllers and data processors. Since the PLA outline is originally based on the former EU data protection regulation (*Directive 95/46/EC* [197]), we update the current PLA outline by comparing the GDPR with the *Directive 95/46/EC* (Section 3.4.1).
- We describe the structure of the updated PLA outline by providing a metamodel (Section 3.4.2).

The remainder of this chapter is organized as follows. In Section 3.2, we present the necessary background for this chapter. In Section 3.3, the structure of privacy preferences is defined. In Section 3.4, we present formalized *PLA*s. In Section 3.5, we discuss our proposed concepts and future work. In Section 3.6, we discuss related work. In Section 3.7, we conclude.

3.2 Background

We first introduce the four key elements of privacy, which are used to define the privacy preferences. We further provide the required basics on the theory of sets and lattices, which are used in our definition of privacy preferences. Eventually, we introduce the *PLA* outline.

3.2.1 The Four Key Elements of Privacy

Barker et al. [22] discussed several definitions of privacy [2, 31, 199] and provided a data privacy taxonomy that is capable of considering privacy technologically. They showed the applicability of their taxonomy in several different real-world settings. Furthermore, they tested the taxonomy against a wide-range of existing work. Barker et al.' taxonomy of privacy includes four key privacy elements, namely *purpose*, *visibility*, *granularity*, and *retention*. In this thesis, due to the correspondence between these elements and the principles relating to the processing of personal data (Article 5 of the GDPR), we use the four key elements of privacy to establish the privacy preferences of a piece of personal data.

The key privacy elements:

- **Purpose** is the basic element of data privacy. It indicates the authorized reasons to process personal data [90]. Data owners have different incentives for providing data to a service provider, i.e., they release data for specific purposes. Therefore, it is mandatory to explicitly identify and collect the purposes, for which a piece of data is released, and verify whether the data processing conforms to the collected purposes.
- **Visibility** controls the number and the kind of users who can process the data for a specific purpose with respect to an operation [90]. Data visibility may include a data owner—who provides data—various users and resources that process data. In this thesis, we use the term *subject* for various data users,

including a natural person, a department, an organization, or any resource that may process data.

- **Granularity** refers to the characteristics of data that could be used to facilitate the proper use of the data, where there exist different valid accesses for various purposes [22]. In other words, data granularity specifies how much precision is provided when presenting data in response to a legitimate query. This is particularly important when a query originates from a different service provider [90].
- **Retention** refers to the need for restricting access or removing the personal data after they have been processed for the authorized purposes.

Concerning these definitions, *purpose* is the most fundamental element and the rest (*visibility, granularity*, and *retention*) are defined in regard to the purpose.

3.2.2 A Background on the Theory of Sets and Lattices

This section is mainly based on the theorems and the models of set theory and lattices provided in [123, 150].

As a foundation for the concept of lattice, we start with the definition of partially ordered set (poset).

Definition 3.2.1. ***Partially Ordered Set:*** *A partially ordered set (poset) $P = (P, \leq)$ is a nonempty set with the binary relation $\leq$ on P satisfying for all $x, y, z \in P$,*

$$\begin{aligned} &(1)\ x \leq x && (\textit{reflexivity}) \\ &(2)\ \textit{if } x \leq y \textit{ and } y \leq x,\ \textit{then } x = y && (\textit{antisymmetry}) \\ &(3)\ \textit{if } x \leq y \textit{ and } y \leq z,\ \textit{then } x \leq z && (\textit{transitivity}) \end{aligned}$$

One of the most natural examples of a partially ordered set is the set of all subsets of a set (powerset), ordered by *inclusion* ($\subseteq$) [150]. In a poset not every pair of elements needs to be comparable (concerning the relation). The set $\Re$ of real numbers with its natural order is another example of a poset. However, $\Re$ is a special type of poset, namely a *totally ordered set*, or *chain*.

Definition 3.2.2. ***Totally Ordered Set:*** *A totally ordered set (chain) is a special type of partially ordered set. $(C, \leq)$ is a chain if for every $x, y \in C$, either $x \leq y$ or $y \leq x$.*

In chain C, every pair of elements are comparable. For every $x, y \in C$, if $x < y$ and there is no $z \in P$ where $x < z < y$, then x is *covered* by y (written $x \prec y$).

Definition 3.2.3. *If $P = (P, \leq)$ is a partially ordered set, and S is a nonempty subset of P, and element $a \in P$, then:*

> *a is an* upper bound *of S if $(\forall x \in S)$ $x \leq a$,*
>
> *a is a* lower bound *of S if $(\forall x \in S)$ $a \leq x$,*
>
> *a is the* least upper bound *(l.u.b.) of S, if a is an* upper bound *of S, and $a \leq y$ for every upper bound y of S,*
>
> *a is the* greatest lower bound *(g.l.b.) of S, if a is a* lower bound *of S, and $y \leq a$ for every lower bound y of S.*

Concerning the definitions above, a lattice is defined as follows:

Definition 3.2.4. ***Lattice:*** *A partially ordered set $(L, \leq)$ is a lattice if for every two elements a and b ($a, b \in L$) the least upper bound (called join, denoted by ($a \vee b$)) and greatest lower bound (called meet, denoted by ($a \wedge b$)) exist.*

Two *lattice-like* structures that arise from the definition of a lattice are:

Definition 3.2.5. ***Join-Semilattice:*** *A partially ordered set $(L, \leq)$ is called a join-semilattice if for every two elements a and b ($a, b \in L$) the least upper bound exists.*

Definition 3.2.6. ***Meet-Semilattice:*** *A partially ordered set $(L, \leq)$ is called a meet-semilattice if for every two elements a and b ($a, b \in L$) the greatest lower bound exists.*

As a matter of convention, the operation symbol of a join-semilattice is denoted by $\vee$ and the operation symbol of a meet-semilattice is denoted by $\wedge$. A lattice $(L, \wedge, \vee)$ is both a join-semilattice and meet-semilattice.

Definition 3.2.7. ***Sublattice:*** *A nonempty subset S of a lattice L is a sublattice of L, where for every pair of elements $a,\ b\ \in S$, both $a \vee b$ (join) and $a \wedge b$ (meet) are in S.*

Concerning the definition of a sublattice, an *interval sublattice* is a special sublattice that denotes a certain quotient (a finite set of elements) of a lattice.

Definition 3.2.8. ***Interval Sublattice:*** *Let $(L, \leq)$ be a lattice and a, b, and x be three member elements of $(L, \leq)$ $(a, b, x \in L)$. Three interval sublattices a/b, $a/g.l.b.(L)$, and $l.u.b.(L)/a$ are defined as:*

$$a/b = \{x \in L \;:\; b \leq x \leq a\}$$
$$a/g.l.b.(L) = \{x \in L \;:\; x \leq a\}$$
$$l.u.b.(L)/a = \{x \in L \;:\; a \leq x\}$$

In this thesis, the key privacy elements (*purpose, visibility*, *granularity*, *retention*) are structured in *lattice* structures.

3.2.3 Privacy Level Agreements

Privacy level agreement (PLA) outline [49] is originally introduced by the Cloud Security Alliance. It is intended to be an appendix to a service level agreement (SLA) and to describe the level of personal data protection provided by service providers to service customers in a structured way. While SLAs generally provide metrics and necessary information on the performance of the services, PLAs address information privacy and personal data protection practices. The first version of the PLA ([V1])—released in 2013—[48] was based not only on the EU personal data protection legal requirements but also on a set of best practices and recommendations. The second version of the PLA ([V2])—released in 2015—is only based on the EU personal data protection legal requirements.

The PLA [V2] outline comprises different paragraphs. It starts by providing a set of generic information on service providers (data processors), such as the address and the relevant data protection officer. Afterwards, mainly, the processing of personal data is described. It indicates the purpose of the processing and the necessary legal basis to perform processing. Furthermore, it specifies if data are transferred to other controllers or processors and which technical security and privacy controls are in place to support privacy and security concerns. There is a paragraph on the data retention policies and the conditions for restricting or deleting data. Eventually, the policies and procedures to ensure and demonstrate compliance by the service provider are described (accountability) and the cooperation with customers to ensure compliance with applicable data protection provision is specified.

In this thesis, we leverage the PLA [V2] outline to establish agreements between data controllers and data processors to systematically capture the privacy preferences and provide a baseline to perform a privacy analysis.

3.3 Privacy Preferences

Article 5 of the GDPR stipulates six principles for the processing of personal data. Personal data shall be:

> "(a) Processed lawfully, fairly and in a transparent manner in relation to the data subject [...];"
>
> "(b) Collected for specified, explicit and legitimate purposes and not further processed in a manner that is incompatible with those purposes (*purpose*) [...];"
>
> "(c) Adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed (*purpose*);"
>
> "(d) Accurate and, where necessary, kept up to date (*granularity*) [...];"
>
> "(e) Kept in a form which permits identification of data subjects for no longer than is necessary for the purposes for which the personal data are processed (*visibility, retention*) [...];"
>
> "(f) Processed in a manner that ensures appropriate security of the personal data [...]."

These principles correspond to the *key elements of privacy* introduced in Barker et al.'s seminal taxonomy (Section 3.2.1): *purpose*, *visibility*, *granularity*, and *retention*. Above, for each principle, the relevant key privacy element(s) is specified (in parentheses). The principle (a) is rather general, however, the fairness requirement of a data processing may be subject to an analysis [174]. The principle (f) corresponds to security requirements (confidentiality, integrity).

In this thesis, we define the privacy preferences based on the four key privacy elements. For each piece of personal data *pd*, it has to be specified:

- For which authorized purpose(s), *pd* may be processed.
- Who is allowed to process *pd* for the authorized purposes.
- With how much precision, *pd* may be transferred to other data processors or data controllers for the authorized purposes.
- When *pd* has to be removed or restricted after it has been used for the authorized purposes.

Consequently, the privacy preferences of a piece of personal data pd are indicated by a set of purposes P' ($P' \subseteq PS$, where PS indicates the set of all possible processing purposes). Furthermore, for each purpose $p \in P'$, the following elements have to be defined:

- a set of subjects V' ($V' \subseteq VS$, where VS indicates the set of all available subjects who may potentially process pd).
- a granularity level $g \in GS$ (GS indicates the set of all possible granularity levels).
- a retention condition $r \in RS$ (RS indicates the set of all possible retention conditions).

Earlier work [17, 82, 173, 180, 202–204] suggest that many privacy-related attributes such as the key elements of privacy (the sets PS, VS, GS, and RS) are best arranged in hierarchical structures such as lattices. Such organized structures simplify the specification of the privacy preferences and the process of privacy analysis. In our work (following the work introduced in [90, 202, 204]), we consider lattices to arrange the purpose, visibility, granularity, and retention sets. In each lattice, the sets of the key elements are organized from a most general, least upper bound, to a most specific, greatest lower bound.

Definition 3.3.1. ***Purpose Lattice:*** *Let PS (*purpose set*) be a partially ordered set of purposes for which personal data are processed. $(PL, \leq)$ is a* purpose lattice *if $\forall\ a, b \in PL$:*

$$(1)\ a, b \in PS,$$
$$(2)\ a \vee b = l.u.b.(a, b) \text{ and } a \wedge b = g.l.b.(a, b).$$

The first property indicates that every comparable pair in a PL is a member of the corresponding PS. The second property indicates every pair of elements has a greatest lower bound and a least upper bound.

A *purpose lattice* organizes all the purposes defined to process personal data in a lattice structure from the general purposes to specific purposes. As mentioned earlier, a piece of personal data may be processed for several purposes. Hence, there exists a set containing all possible purposes to process a piece of personal data. A number of purposes in this set are comparable—there is a relation between them. A purpose may relate to other purpose(s), in a way that it is more specific than the other(s). For instance, we may have two purposes, namely *marketing* and *societal marketing*[1]. In this case, the *societal marketing* is more specific than *marketing*, i.e.

[1] The societal marketing is a marketing concept, where the marketing decisions are not only based on the consumer's data, but also considers the society's long-term interests [100].

societal marketing is a special type of *marketing*. Therefore, we may say that the set of all purposes to process a piece of personal data is a *partially ordered set*, where not every pair of elements need to be comparable. Furthermore, in this set, we consider one purpose as the most general purpose and one purpose as the most specific purpose. The most general purpose is the most intuitive purpose to process a piece of personal data. The most specific purpose is an ultimate purpose, thereby enabling an official authority to process a piece of personal data without any restriction.

Concerning the fact that the set of all possible purposes to process a piece of personal data is a partially ordered set with a most general and most specific purposes, we represent this set as a lattice.

Definition 3.3.2. ***Subsume Relation (purpose lattice):*** *Let* $(PL, \leq)$ *be a purpose lattice. Given purposes* $a,\ b \in PL$*, if b is a parent purpose of a (there is an upward path from* a *to b) then purpose* a subsumes *purpose b (*$a \geq_s b$*).*

The least upper bound ($l.u.b.$) of a PL is the most general (intuitive) purpose to process a piece of personal data. This purpose is subsumed by all other purposes. The greatest lower bound ($g.l.b.$) of a PL is the most specific (extreme) purpose to process a piece of personal data. This purpose subsumes all other purposes.

In Figure 3.2, concerning our example scenario (Section 2.2), we present a sample purpose lattice, where *issue* is the most general purpose to process *social security number* (*SSN*), and *legal request* is the most specific purpose. In this purpose lattice, *societal marketing* subsumes *marketing*. In reality, a purpose lattice includes more purposes, however, we only show the significant purposes. For better understanding, generally, we simplified the lattices in this thesis.

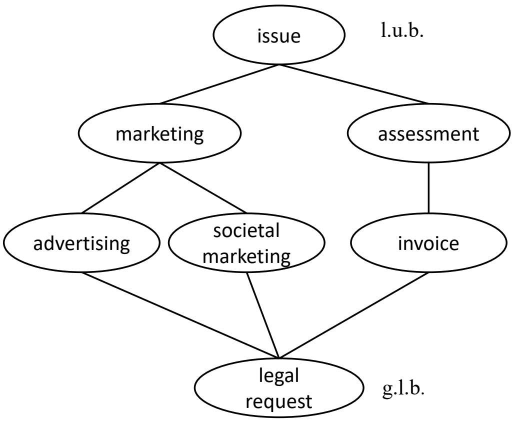


Figure 3.2: A sample *purpose lattice*

Definition 3.3.3. ***Visibility Lattice:*** *Let VS (*visibility set*) be a partially ordered set of all subjects and resources who may process personal data.* ($VL, \leq$) *is a* visibility lattice *if* $\forall\, a, b \in VL$*:*

$$(1)\ a, b \in VS,$$
$$(2)\ a \vee b = l.u.b.(a, b) \text{ and } a \wedge b = g.l.b.(a, b).$$

The first property indicates that every comparable pair in a VL is a member of the corresponding VS. The second property indicates every pair of elements has a greatest lower bound and a least upper bound.

A piece of personal data may be processed by several subjects. A *visibility lattice* organizes all subjects who exist and may process personal data. The least upper bound ($l.u.b$) of a VL is the most general subject who may process a piece of personal data. In this thesis, similar to the earlier work [202], we consider the data owner as the most general subject who may process a piece of personal data. The most specific subject (greatest lower bound of a VL) who may process a piece of personal data may account for the whole world or a powerful authority with unrestricted rights to access personal data. The subsume relation (Definition 3.3.2) applies to a visibility lattice as well, where a more specific subject subsumes a general subject.

Figure 3.3 illustrates a sample visibility lattice, where an *owner* (a data provider or data subject) is the least upper bound and the *world* is the greatest lower bound. Furthermore, different departments are shown in the visibility lattice, which may process a piece of personal data. Concerning the subsume relations in a visibility lattice, if the *finance department* is authorized to process a piece of personal data *pd*, the *business department* is authorized to process *pd* as well.

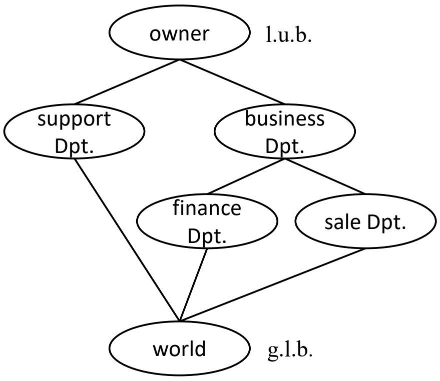


Figure 3.3: A sample *visibility lattice*

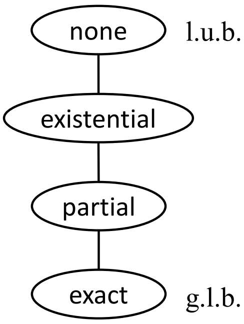


Figure 3.4: A *chain* showing the four granularity levels

Definition 3.3.4. ***Granularity Lattice:*** *Let GS (*granularity set*) be a partially ordered set of all existing granularity levels to process personal data.* $(GL, \leq)$ *is a* granularity lattice *if* $\forall\, a, b \in$ *GL:*

$$(1)\ a, b \in GS,$$
$$(2)\ a \vee b = l.u.b.(a, b) \text{ and } a \wedge b = g.l.b.(a, b).$$

The first property indicates that every comparable pair in a GL is a member of the corresponding GS. The second property indicates every pair of elements has a greatest lower bound and a least upper bound.

A *granularity lattice* organizes all possible granularity levels with various precision into a lattice structure from the most general granularity level *none* (providing no personal data) to the most specific (precise) granularity level *exact*. The least upper bound does not disclose any data, while the greatest lower bound discloses the exact data value. Depending on the data type, different granularity levels may be defined. In this work, we consider only four granularity levels *none, existential, partial*, and *exact*. The *existential* granularity level only specifies if a piece of personal data exists. The *partial* granularity level reveals data only to a limited extent and not entirely. Since a GL only contains four levels with a subsume relation between each two levels (any pair of elements are comparable), the granularity lattice is, in fact, a *chain* (*total order*), represented as a linear structure (Figure 3.4).

Definition 3.3.5. ***Retention Lattice:*** *Let RS (*retention set*) be a partially ordered set of all existing retention conditions to process personal data.* $(RL, \leq)$ *is a* retention lattice *if* $\forall\, a, b \in$ *RL:*

$$(1)\ a, b \in RS,$$
$$(2)\ a \vee b = l.u.b.(a, b) \text{ and } a \wedge b = g.l.b.(a, b).$$

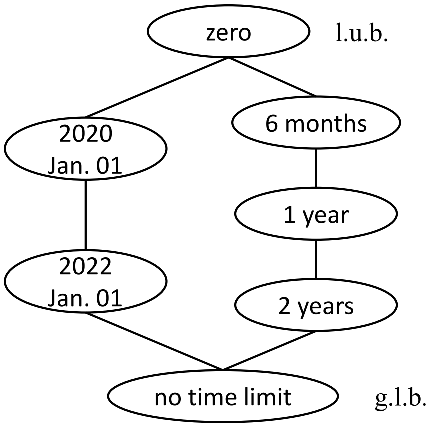


Figure 3.5: A sample *retention lattice*

The first property indicates that every comparable pair in a RL is a member of the corresponding RS. The second property indicates every pair of elements has a greatest lower bound and a least upper bound.

A *retention lattice* specifies all possible retention conditions. The retention conditions subsume each other. The lowest upper bound may be defined with *zero* prohibiting a piece of personal data from storage or processing. The greatest lower bound may be specified by *no time limit*. Between these two conditions, various retention conditions are defined. In Figure 3.5, we show a sample retention lattice.

After defining the four lattices, we now provide a formal definition for the privacy preferences:

Definition 3.3.6. ***Privacy Preferences:*** *Let PL be a purpose lattice, VS be a visibility set, GL be a granularity lattice, and RL be a retention lattice. The privacy preferences (PrP) of a piece of personal data pd is defined:*

$$PrP_{pd} = PL \nrightarrow \mathcal{P}(VS) \times GL \times RL$$

Where:

1. *$\nrightarrow$ is a partial function.*
2. *$\mathcal{P}(VS)$ is the power set of the set VS.*

In a nutshell, the privacy preferences of a piece of personal data pd are defined as a set of purposes—for which pd is authorized to be processed—where each purpose in this set is mapped to other three key privacy elements. According to Definition 3.3.6, the mapping from PL to $\mathcal{P}(VS) \times GL \times RL$ is denoted by a *partial function*

($\nrightarrow$), indicating the fact that not for every purpose of a purpose lattice, a mapping is required, i.e., not every purpose that belongs to a purpose lattice may be included in privacy preferences.

Furthermore, in this definition, instead of *VL*, $\mathcal{P}(VS)$ is used, indicating that each authorized purpose p may be mapped to several subjects, who are authorized to process a piece of personal data for purpose p. A power set $\mathcal{P}(VS)$ (the set of all subsets of *VS*) can be represented as a lattice with *inclusion* relation. However, the subsume relation cannot be derived from the lattice that illustrates $\mathcal{P}(VS)$. For instance, consider the partially ordered set $V = \{v_1, v_2, v_3, v_4\}$, where v_1 is the most general subject (*l.u.b*), v_4 is the most specific subject (*g.l.b*), v_2 and v_3 are both more specific than v_1 and more general than v_4 (concerning the subsume relation: $v_1 \leq_s v_2$, $v_1 \leq_s v_3$, $v_2 \leq_s v_4$, $v_3 \leq_s v_4$). In Figure 3.6, the left lattice illustrates the visibility lattice $(VL, \leq)$. The right lattice illustrates the powerset of V ($\mathcal{P}(V)$). Given the set V (introduced above), $p \in PL$, $g \in GL$, and $r \in RL$, for a piece of personal data *pd*, the following privacy preferences may be defined:

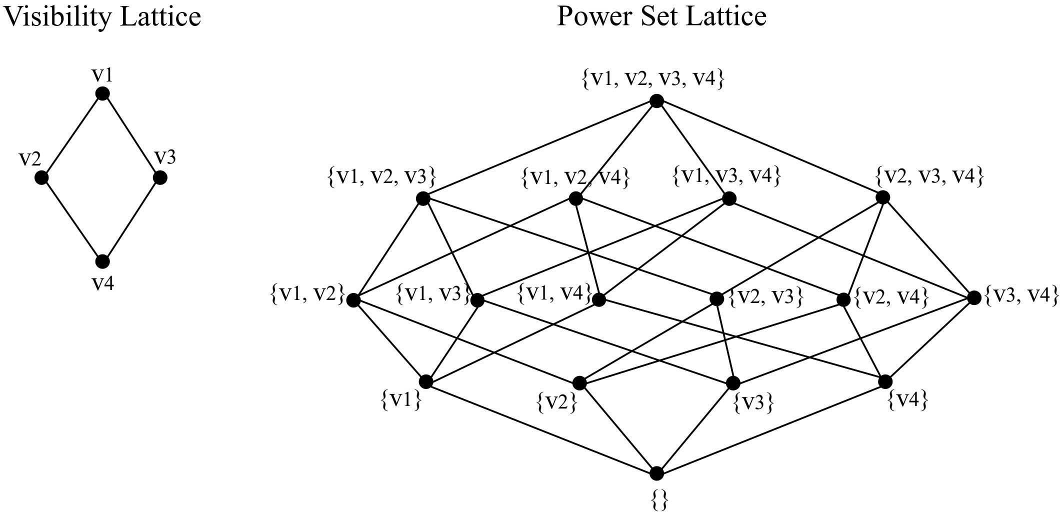


Figure 3.6: The left lattice illustrates the visibility lattice $(VL, \leq)$. The right lattice shows the powerset of V ($\mathcal{P}(V)$).

> "v_2, and v_3 are authorized to process *pd* for purpose p, granularity level g, and retention time r."

Obviously, purpose p is mapped to $\{v_2, v_3\}$, where $\{v_2, v_3\} \in \mathcal{P}(V)$. Concerning the definition of set V, v_2 and v_3 subsume v_1. Since v_2 and v_3 are authorized to process *pd*, v_1 is authorized to process *pd* as well. This cannot be derived from the powerset lattice but from the visibility lattice. Therefore, in Definition 3.3.6, a

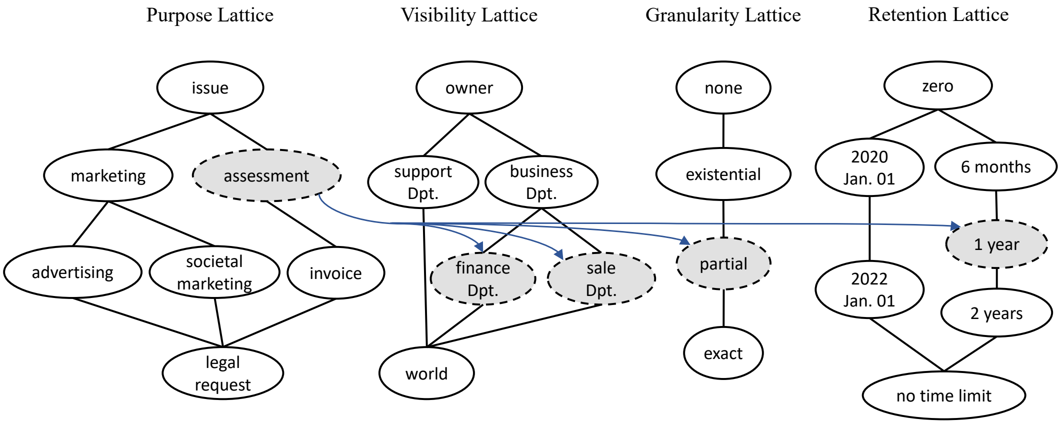


Figure 3.7: An example of the privacy preferences of *SSN*. The dashed lines indicate the privacy preferences (authorized purposes, subjects, granularity levels and retention conditions).

purpose is mapped to a subset of VS, however, to identify who is authorized to process a piece of personal data the visibility lattice is required.

In Section 2.2, we introduced our example scenario, where a piece of personal data *SSN* (*Social Security Number*) is processed. We may consider the following privacy preferences for *SSN*:

> *SSN* is only authorized to be processed for the *assessment* purpose. The *finance* and *sale department*s are authorized to process *SSN* for the *assessment* purpose. The authorized precision level to process *SSN* is *partial* and *SSN* has to be removed or restricted within one year after it has been processed for the *assessment* purpose.

We use the following notation to show the privacy preferences of *SSN* (PrP_{SSN}):

$$PrP_{SSN} = \{(assessment \mapsto (\{financeDept, saleDept\}, partial, 1year))\}$$

In this notation, $\mapsto$ is used to express that an authorized purpose is the central privacy element and the other three elements are mapped to the authorized purpose. Figure 3.7 represents the above-mentioned privacy preferences in the corresponding lattices.

Concerning the set PrP_{SSN} given above, the privacy preferences of *SSN* includes only one element (a mapping between a purpose and the other three elements).

However, the PrP_{SSN} may include several elements. The set of privacy preferences of a piece of personal data pd is a set $PrP_{pd} = \{b_1, b_2, ..., b_n\}$, where each b intrinsically (by Definition 3.3.6) is shown as $b = p \mapsto (VS', g, r)$, where $p \in PL$, $VS' \subseteq VS$, $g \in GL$, $r \in RL$.

Concerning the PrP_{SSN}, of special importance is the consideration of the subsume relation in the lattices. According to the nature of a purpose lattice, if in the privacy preferences of a piece of personal data *pd*, a purpose $p \in PL$ is defined as an authorized purpose, all the purposes that belong to PL and are subsumed by purpose p are authorized purposes as well. The set $PrP_{pd} = \{b_1, b_2, ..., b_n\}$ in fact denotes a set of *base* privacy preferences $Base\text{-}PrP_{pd}$. Concerning the nature of lattices, the base privacy preferences account for a set of *effective* privacy preferences, indicated as:

$$\begin{aligned}
Eff\text{-}PrP_{pd}(Base) = &\{(p, VS', g, r) \mid p \in l.u.b.(PL)/p_{b_i} \\
&and\ VS' \subseteq (V_{b_i}, \vee) \\
&and\ g \in l.u.b.(GL)/g_{b_i} \\
&and\ r \in l.u.b.(RL)/r_{b_i}, \\
&for\ all\ b_i \in Base\text{-}PrP_{pd} \\
&and\ i \in \{1, ..., n\}\}
\end{aligned}$$

Based on the subsume relation in the lattices, each purpose p_{b_i}, visibility set VS'_{b_i}, granularity level g_{b_i}, and retention condition r_{b_i} in the element b_i of the set $Base\text{-}PrP_{pd}$, account for a set, expressed by an interval sublattice (see Definition 3.2.8) or a join-semilattice (see Definition 3.2.5).

Following the effective privacy preferences of a piece of personal data *pd*, in what follows four lemmas are provided to denote the sets of all authorized key privacy elements for processing *pd*.

Lemma 3.3.1. *Let $PrP_{pd} = \{b_1, b_2, ..., b_n\}$ be the set of base privacy preferences of a piece of personal data pd, PL be a purpose lattice, p_{b_i} denotes the authorized purpose specified in the element b_i of the* base *privacy preferences of pd, then pd is authorized to be processed for the set of purposes $\mathcal{P}_{pd}$ defined as:*

$$\mathcal{P}_{pd} = \bigcup_{i=1}^{n} l.u.b.(PL)/p_{b_i}.$$

Proof. In a purpose lattice PL, each purpose p, specified in the privacy preferences of a piece of personal data pd, subsumes a set of purposes—if p is not a least upper bound ($l.u.b.$), necessarily it subsumes at least the $l.u.b.$—thereby indicating the fact that pd is authorized to be processed for the subsumed purposes as well.

$l.u.b(PL)/p = \{x \in PL \mid x \leq_s p\}$ is an interval sublattice which denotes a set including purpose p and all the purposes that are subsumed by p. Therefore, in fact, the interval sublattice $l.u.b.(PL)/p_{b_i}$ specifies the set of authorized purposes for which pd can be processed, concerning the element b_i. The set of all purposes for which pd is authorized to be processed (denoted by $\mathcal{P}_{pd}$) is indicated by the union of all stated interval sublattices $\bigcup_{i=1}^{n} l.u.b.(PL)/p_{b_i}$. □

Lemma 3.3.2. *Let $PrP_{pd} = \{b_1, b_2, ..., b_n\}$ be the set of base privacy preferences of a piece of personal data pd, VS be the set of all subjects who may process pd, purpose p_{b_i} be an authorized purpose specified in the element b_i of the* base *privacy preferences, and VS'_{b_i} ($VS'_{b_i} \in \mathcal{P}(VS)$) be the set of subjects, to which p_{b_i} is mapped, then the set of subjects who are authorized to process pd for purpose p_{b_i} is denoted by the join-semilattice:*

$$(V_{b_i}, \vee)$$

Proof. Since the cardinality of VS'_{b_i} ($|VS'_{b_i}|$) may be greater than one, an interval sublattice (similar to lemma 3.3.1) cannot represent the set of all authorized subjects. VS'_{b_i} may have several elements and each element represents an interval sublattice. The authorized subjects are denoted by a join-semilattice (union of several interval sublattices). According to Definition 3.2.5, in a join-semilattice $(V, \vee) \subseteq VL$, every two elements have a least upper bound (*data owner*).

The join-semilattice $(V_{b_i}, \vee)$, where V_{b_i} is equal to the set VS'_{b_i}—to which p_{b_i} is mapped—of the element b_i of the *base* privacy preferences of pd, specifies the set of subjects who are authorized to process pd for purpose p_{b_i}. □

A join-semilattice $(V_{b_i}, \vee)$ has a set of lower bounds $V^l_{b_i}$, which is equal to the set VS'_{b_i} and contains the specific subjects that subsume the other subjects included in the join-semilattice $(V_{b_i}, \vee)$.

$$V^l_{b_i} = \{x \in V_{b_i} \mid x \geq_s v, \ \forall v \in V_{b_i}\}$$

As a result, the set of subjects who are authorized to process pd for the purpose p_{b_i}, alternatively may be denoted by the set of lower bounds $V^l_{b_i}$.

Lemma 3.3.3. *Let $PrP_{pd} = \{b_1, b_2, ..., b_n\}$ be the set of base privacy preferences of a piece of personal data pd, GL be a granularity lattice, purpose p_{b_i} be an authorized purpose specified in the element b_i of the* base *privacy preferences, and g_{b_i} be the granularity level, to which p_{b_i} is mapped, then the set of granularity levels, with which pd is authorized to be processed for purpose p_{b_i}, is denoted by:*

$$l.u.b.(GL)/g_{b_i}$$

Proof. Likewise Lemma 3.3.1, each $g \in GL$ indicates an interval sublattice $l.u.b.(GL)/g$. Therefore, the interval sublattice $l.u.b.(GL)/g_{b_i}$, indicates the granularity levels with which pd is authorized to be processed for purpose p_{b_i}. □

As it is mentioned earlier, in this work, only a simple granularity lattice represented as a linear structure (chain) is considered. Hence, $l.u.b.(GL)/g_{b_i}$ is always a linear interval sublattice.

Lemma 3.3.4. *Let $PrP_{pd} = \{b_1, b_2, ..., b_n\}$ be the set of base privacy preferences of a piece of personal data pd, RL be a retention lattice, purpose p_{b_i} be an authorized purpose specified in the element b_i of the* base *privacy preferences, and r_{b_i} be the retention condition, to which p_{b_i} is mapped, then the set of retention conditions, which specifies when pd has to be removed or restricted after it has been used for the authorized purpose p_{b_i}, is denoted by:*

$$l.u.b.(RL)/r_{b_i}$$

Proof. Likewise Lemma 3.3.1, each $r \in RL$ indicates an interval sublattice $l.u.b.(RL)/r$. Therefore the interval sublattice $l.u.b.(RL)/r_{b_i}$, indicates the retention conditions specified in the element b_i of the *base* privacy preferences of a piece of personal data pd. □

Considering Lemmas 3.3.1-3.3.4, the lattices in Figure 3.7 and the privacy preferences of the *SSN* specified by:

$$PrP_{SSN} = \{(assessment \mapsto (\{financeDept, saleDept\}, partial, 1year))\}$$

Since *assessment* subsumes *issue* in the purpose lattice, *SSN* is authorized to be processed for the *issue* purpose as well. The *assessment* purpose is mapped to the set

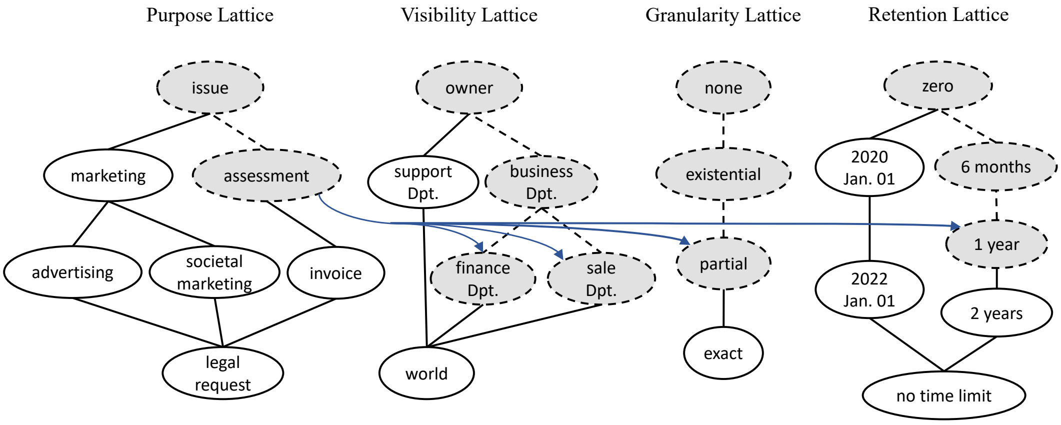


Figure 3.8: The effective privacy preferences of *SSN*. The dashed lines indicate the privacy preferences (authorized purposes, subjects, granularity levels, and retention conditions).

$\{financeDept, saleDept\}$, indicating that the *finance* and *sale departments* are authorized to process *SSN* for the *assessment* purpose. These two subjects are the lower bounds of the join-semilattice that includes the subjects that are authorized to process *SSN* for the *assessment* purpose. Furthermore, *SSN* is authorized to be processed for the *existential* precision level as well, and *SSN* may be removed or restricted within any time period less than one year.

Figure 3.8 represents the *effective* privacy preferences of *SSN*. The *base* privacy preferences ($\{(assessment \mapsto (\{financeDept, saleDept\}, partial, 1year))\}$) accounts for 270 *effective* privacy preferences ($2 \times (2^4 - 1) \times 3 \times 3$). Some of the *effective* privacy preferences of *SSN* (*Eff*-PrP_{SSN}) are:

$$\begin{aligned} \textit{Eff-}PrP_{SSN} = \{&(issue \mapsto (\{financeDept, saleDept\}, partial, 1year)),\\ &(issue \mapsto (\{businessDpt\}, existential, 1year)),\\ &(assessment \mapsto (\{businessDpt\}, partial, 1year)),\\ &(assessment \mapsto (\{financeDept\}, existential, 1year)), ...\} \end{aligned}$$

The total number of the elements of *Eff*-PrP_{SSN} is obtained by: $|l.u.b.(PL)/p| \times (2^{|(V,\vee)|} - 1) \times |l.u.b.(GL)/g| \times |l.u.b.(RL)/r|$. For the *PL*, *GL*, and *RL*, the cardinality of the corresponding interval sublattices are considered. Concerning *VL*, since a purpose is mapped to a set, which may include more than a subject, the cardinality of the powerset of the join-semilattice ($2^{|(V,\vee)|}$) is considered. However, since the empty set is a member of every powerset of a set and we do not consider the empty set in the visibility lattice, the cardinality of the powerset is subtracted by one.

3.4 Formalized Privacy Level Agreements

We previously motivated the need for a structured means to specify the privacy preferences of personal data determined by a data controller (service customer). To this end, we propose to use the PLA outline [49] (see Section 3.2.3) to establish agreements on the use of personal data between data controllers and data processors, thereby expressing the privacy preferences.

The PLA outline is only provided in a textual format. Therefore, in this section, we present a metamodel to systematically specify the structure of a PLA. The PLA outline [49] is heavily based on Directive 95/46/EC [197] (the former EU data protection regulation). Thus, before introducing the PLA metamodel, we compare the GDPR with *Directive 95/46/EC* to update and extend the PLA outline and ensure its compliance with the GDPR.

3.4.1 A Brief Description of the Differences Between the GDPR and Directive 95/46/EC

The GDPR [198] repeals Directive 95/46/EC, thereby updating and modernizing the principles stated in the Directive 95/46/EC to guarantee privacy rights. According to the European Commission, the GDPR focuses on: reinforcing individuals' rights, strengthening the EU internal market, ensuring stronger enforcement of the rules, streamlining international transfers of personal data, and setting global protection standards. The GDPR provides the data subjects more control over their personal data [75].

The GDPR adds some new definitions, updates some of the basic principles, and formulates some new principles. From the 99 articles contained in the GDPR, 26 articles are not directly mentioned or contained in *Directive 95/46/EC*. 37 articles are updated or are described more comprehensively. In this section, we do not aim to completely compare the GDPR with Directive 95/46/EC. We only highlight the updated/added principles that are essential to present the PLA metamodel and are required as the basis for the rest of this work. Our comparison is documented in an Excel file[2]. An excerpt of this comparison is provided in Appendix B.

Following our comparison: In Article 4 of the GDPR, new definitions such as genetic and biometric data, profiling and pseudonymization are added. Similarly, Article 9 of the GDPR defines new special categories of personal data. The new definitions have to be supported in the PLA outline. In Article 5 of the GDPR, the

[2] `https://cloud.uni-koblenz-landau.de/s/ocRXY9nJqDWzgpA` (accessed: 2019-06-01)

principles relating to the processing of personal data remain unchanged, however, new concepts such as accuracy, data minimization, purpose limitation, storage limitation are explicitly stated.

Directive 95/46/EC defines the data subject's consent and in Articles 7 and 8, it prescribes that for the processing of personal data and special categories of personal data, the data subject has to give her/his consent unambiguously. These are stipulated in the GDPR as well. Additionally, the GDPR prescribes a set of conditions for consent and (precisely) a child's consent. In this work, indeed by expressing the privacy preferences in the PLAs, we declare the consents. Since in the PLA outline, the consent is not directly mentioned, the outline should be accordingly adjusted to be consistent with the GDPR. Directive 95/46/EC does not prescribe data portability. The GDPR describes the right to data portability in Article 20. Data portability is supported in Paragraph 7 of the PLA outline.

Article 25 of the GDPR prescribes *privacy by design* and Article 35 prescribes *privacy impact assessment*. Although these concepts already exist, they were not legally binding before, i.e., they were not stated explicitly in the Directive 95/46/EC. The PLA outline has to support them.

Concerning our discussion in this section, in Appendix C, we propose an extension (update) of the current PLA outline.

3.4.2 The PLA Metamodel

The main aim of formalizing the PLA outline is to capture privacy preferences. An agreement is later used as input to a privacy analysis to verify whether the specified privacy preferences are supported by a system.

Figure 3.9 demonstrates a metamodel for the PLA outline. A *data controller* determines the purposes and means of processing a piece of *personal data* by specifying a set of *privacy preference*s. The personal data is processed by a *data processor*. A data processor conducts a set of *processes*. Each process is realized by a set of *operation*s to process a piece of personal data. Pursuant to the PLA outline and the GDPR, for each operation (process), the purpose of processing has to be defined. This is specified by the *Objective* class.

According to Definition 3.3.6, the privacy preferences are based on the four key privacy elements. Since purpose is the key element (see Section 3.2.1), in the metamodel, the other three privacy elements are defined in regard to a purpose. Concerning the privacy preference class and it's associated classes in the metamodel,

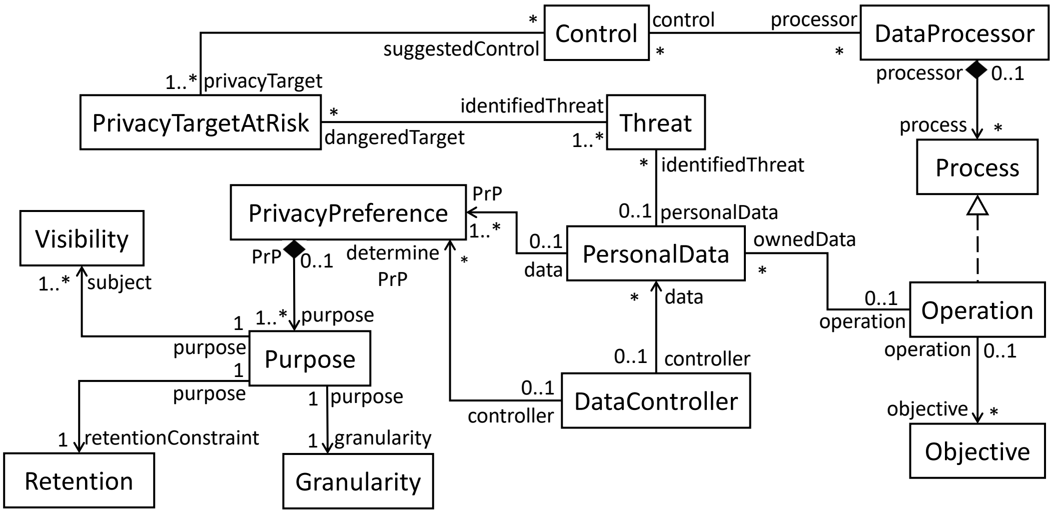


Figure 3.9: The PLA metamodel

the multiplicities of the associations indicate that:

- For each piece of personal data, at least one specific *purpose* for processing has to be determined.
- For each specific purpose:
  - A set of subjects (one or more) has to be specified as authorized subject(s) to access the piece of personal data (*visibility*).
  - A *granularity* level for the piece of personal data, in case it is transferred to other data processors or controllers, has to be specified.
  - A *retention* constraint has to be specified.

We further demonstrate the associated classes of privacy risks in Figure 3.9, namely *Threat, PrivacyTargetAtRisk* and *Control*. The privacy threats arise when processing a piece of personal data without taking into account the privacy preferences. The threats endanger the privacy targets and cause risks. To mitigate the privacy risks, a set of controls may be suggested.

Comparing our metamodel and the PLA outline, a PLA has to include several other classes. For instance, every controller and processor have to designate a *data protection officer*. The processing of personal data has to be monitored by a public authority. Such principles and concepts are out of scope of this thesis and do not appear in the PLA metamodel.

3.5 Discussion and Limitations

In this section, we revisit the research questions *RQ1* and *RQ2*. Afterwards, we discuss the limitations of the concepts proposed in this chapter. We further present the potential extensions and highlight future work.

3.5.1 Revisiting the Research Questions

We explain how the research questions introduced at the beginning of this chapter are addressed.

RQ1: How can privacy preferences be defined? We defined the privacy preferences of personal data relying on the four key privacy elements, namely *purpose*, *visibility*, *granularity*, and *retention*. The four key privacy elements correspond to the principles relating to the processing of personal data (Article 5 of the GDPR). The four sets of all possible key privacy elements are structured in four lattices. The lattices (Definition 3.3.1-3.3.5) and the subsume relations inside the lattices (Definition 3.3.2) enabled a formal definition of privacy preferences. We showed how a sample *base* privacy preferences of a piece of personal data (*SSN*) accounts for 270 *effective* privacy preferences, thereby elucidating the efficiency of our formal definition for privacy preferences.

The sets of all authorized key privacy elements for processing a piece of personal data (Lemmas 3.3.1-3.3.4) are later used to verify whether a system design supports the privacy preferences.

RQ2: How can agreements on the use of personal data be established to systematically specify the privacy preferences and support a privacy analysis? We benefit from the PLA outline originally introduced by Cloud Security Alliance to establish agreements on the use of personal data between data controllers and data processors, thereby specifying privacy preferences. Since PLA is based on Directive 95/46/EC (the former data protection regulation of the EU), we compared the GDPR with Directive 95/46/EC to update the PLA outline. We provided a metamodel to specify the structure of PLAs. Our PLA metamodel particularly supports the definition of privacy preferences. A PLA has several other elements to track privacy threats and risks and specify the privacy controls which mitigate the arising risks. In Section 7.2.3, we discuss tool support to generate agreements that correspond to the PLA structure that we introduced in this chapter.

3.5.2 Limitations

Compound purposes. We leveraged the earlier work by Staden et al. on *purpose organization* [202–204], to define the structure of privacy preferences. Staden et. al. further introduce compound purposes, where a compound purpose relies on single purposes and three operations, namely *or* ($+_p$), *and* ($._p$), and *negation* ($\cdot\neg_p$).

- $._p$ specifies that a piece of personal data is authorized to be processed for all the purposes (operands) to which the operator is applied.
- $+_p$ specifies that a piece of personal data is authorized to be processed for either one of the purposes (operands) to which the operator is applied.
- $\cdot\neg_p$ specifies that a piece of personal data is not authorized to be processed for a certain purpose (operand) to which the operator is applied.

In our definition of privacy preferences (Definition 3.3.6), we do not consider compound purposes. In Staden et al.′ work, the access evaluation only relies on purposes and not other key privacy elements. Using the fundamental operations of set theory namely (union ∪, intersection ∩) we may define compound purposes, however in our work, conforming to the GDPR, explicitly each authorized purpose (indicated in the privacy preferences of a piece of personal data) is mapped to a set of subjects, a granularity level, and a retention condition. Therefore in our definition of privacy preferences, we do not support compound purposes.

Furthermore, considering the subsume relation (Definition 3.3.2), in a purpose lattice, a purpose which subsumes two purposes, accounts for both purposes ($._p$ operation). Moreover, since we require that for each piece of personal data, the authorized purposes are explicitly defined; the $+_p$ operation is avoided.

Specific constraints on lattices. Concerning each lattice introduced in Section 3.3, a service provider may require that the privacy preferences are specified within a specific portion (interval sublattice)—a/b where a and b are two elements of the corresponding lattice—of the lattice. Defining such interval lattices enable a service provider to indicate a necessary (pre-defined) range, in which the privacy preferences may be defined. Such ranges are similar to the specification of a *minimal acceptance level* (*MinAL*) and *maximal acceptance level* (*MaxAL*) introduced in [90, 202].

Granularity lattice. In the granularity lattice (Definition 3.3.4), we only consider four precision levels. Therefore in Figure 3.4, the granularity lattice is represented as a *chain* (Definition 3.2.2). However, depending on the type of personal data, we may consider deferent types of granularity level. In this case, a granularity lattice is represented as a lattice with *none* as the least upper bound and *exact* as the greatest lower bound, and several branches in between, where each branch is a chain with different precision levels of a specific (granularity) type.

A consistent terminology for the lattices. In this chapter, we introduced a definition of privacy preferences based on the four key elements of privacy and the idea of structuring them in lattice structures. Such preferences are specified by data controllers for personal data and have to be supported when processing personal data. As mentioned in Section 2.2, a piece of personal data such as the *social security number* (*SSN*) may be processed in an environment by several data processors. One difficulty that becomes evident regarding the definition of privacy preferences, is ensuring a consistent terminology for purpose, visibility, granularity, and retention lattices among different parties. In fact, by establishing agreements on the use of personal data between two parties, we cover this difficulty. In a PLA established between two parties, the privacy preferences of a data controller have to be explicitly specified, and a processor has to support the privacy preferences, when processing personal data.

One other approach to deal with inconsistencies concerning the lattices and support a common terminology between various parties when processing personal data is standardizing a terminology for the sets of all possible purposes, subjects, granularity levels, and retention conditions. Riehle et al. [175] introduce a methodology to automatically annotate process models in regard to a domain ontology [94]. In the future, we plan to adopt this methodology to establish a common understanding of privacy preferences between different parties who process personal data in a certain environment.

3.6 Related Work

We highlight the earlier work on the privacy preferences, privacy agreements, and similar conceptual frameworks and best practices.

Privacy preferences. Several access control mechanisms and authorization languages gained wider traction from the ongoing trend to establish privacy-aware mechanisms. In [41, 153], the authors enhance *Role Base Access Control* (*RBAC*) [181] model to capture purposes. In their work, they organize the purposes in a tree structure.

In [202–204], the authors propose the organization of the purposes in lattice structures to manage the privacy of IT systems. Their work provides a proper foundation to define privacy preferences in this thesis. In our work, similar to a purpose lattice, various subjects, granularity levels, and retention conditions are organized in lattice structures.

Ghazinour et al. [90] propose a lattice-based privacy-aware access control model. Their model does not rely on a conventional access control model such as *RBAC*, where users are committed to various roles, and permissions are assigned to the roles. The permissions are rather granted based on the four privacy elements, namely purpose, visibility, granularity level, and retention condition. Their work does not enable a privacy analysis to identify privacy design violations. The proposed lattices are used to govern the access rights in a system.

In [20], Azraoui et al. propose A-PPL (Accountable PPL), an accountability policy language that represents machine-readable accountability policies. A-PPL extends the PPL language [200] by allowing the definition of new rules on data retention, data location, logging, and notification. PPL (PrimeLife Policy Language) was proposed by the PrimeLife[3] project to express machine-readable privacy policies. Both languages are built upon XACML (eXtensible Access Control Markup Language).

In the P3P[4] (*Platform for Privacy Preferences*) project, a standard machine-readable format for privacy policies is introduced to enable the websites to express their privacy practices and allow the users to be informed of these practices. P3P provides a set of standard purposes and enables an enterprise to define specific purposes. However, the purposes are not organized, and complex purposes are not supported. The *Enterprise Privacy Authorization Language* (*EPAL*) provides a formal language for writing privacy policies to manage data by relying on fine-grained positive and negative authorization rights. They benefit from purposes to determine authorization. In EPAL, purposes are hierarchical elements. A parent node in this hierarchy is a grouping of children nodes.

The above-mentioned works cover several categories of privacy enhancing technologies (PETs)[5], which support privacy principles, for instance, by informing (privacy declaration), giving control over personal data, or restricting (access to) data. We benefit from the basic concepts of a number of these approaches to define privacy preferences. Such technologies are crucial to ensure privacy, however, they do not enable a privacy analysis in the early phases of the system design. Furthermore, in our work, to specify privacy preferences, we particularly support the GDPR, where purpose is the main privacy element and other elements are specified in regard to the purpose.

Formalized representation of the privacy level agreements. In [66], the authors

[3] `http://www.primelife.eu/` (accessed: 2019-06-01)

[4] `https://www.w3.org/P3P/` (accessed: 2019-06-01)

[5] More details on the categories of PETs are provided in Chapter 6.

propose an ontology-based model to represent the information included in the PLAs and automate the process of enforceable policy creation. Moreover, they extend this ontology to create a link between policy elements in the PLA and the actual policies processed by software systems. Their ontology establishes a mapping between high-level policies and low-level policies. Moreover, Benbernou et al. propose a framework [30] for privacy management in web services. They define a privacy agreement, including a privacy policy and data subject preferences between two parties. Their framework supports the lifecycle management of these privacy agreements by defining a set of events occurring in a dynamic environment and a set of actions to adjust the agreements. Additionally, a negotiation protocol to establish proper interactions between the parties is provided. Our approach on formalizing the PLAs establishes agreements on the use of personal data by specifying personal data that are processed and their privacy preferences. The privacy preferences are based on the principles prescribed in the GDPR. The PLAs are later used to analyze a system and to support the *privacy by design* principle. In fact, our approach considers the early phases of system development and does not aim to enforce policies, monitor a system, or preserve agreements.

Conceptual frameworks, guidelines, and best practices. In [117], a framework is provided to support the elicitation and analysis of security requirements from relevant regulations and laws, and develop a system that satisfies these requirements. In [118], the authors propose an approach to assist organizations in selecting proper cloud models by supporting the elicitation and the analysis of their privacy and security needs. In our approach, we provide a structured approach to describe the processing of personal data and specify the privacy preferences in regard to the GDPR.

In [155], Oberholzer et al. introduce the privacy contracts as a means of ensuring data protection by organizations and specifying the privacy preferences by the clients of the organizations. They describe the privacy contract origin and context, the principles on which the privacy contracts are based, and the process of creating and updating the privacy contracts. Their work contributes to the basics of the PLA, however, they do not provide any details on the concrete structure of a privacy level agreement (or a privacy contract)

In [169], a framework for model-based privacy best practice compliance checker that assists the experts to reason about how privacy compliance may be satisfied, optionally using predefined models, is introduced. In their work, they consider top-level security and privacy goals and link them to the system level enforcement technologies. Their approach is generic, and neither demonstrates the processing of personal data, nor the privacy preferences.

3.7 Preliminary Conclusion

In this chapter, we defined privacy preferences. The four key privacy elements, namely purpose, visibility, granularity, and retention constitute the privacy preferences. Furthermore, we proposed to use PLAs to conclude agreements between data controllers and data processors, specifying the privacy preferences of the personal data that are processed. Since the PLA outline (of the CSA) is only provided in a textual format, we formalized a PLA using a metamodel. We particularly included the structure of the privacy preferences in the PLA metamodel. The definition of the privacy preferences and the PLA metamodel adhere to the principles stipulated in the GDPR.

The two concepts, *privacy preferences* and *PLA*, provide the basics for the rest of this thesis. In Chapter 4, we describe the details of performing a privacy analysis with respect to the privacy preferences included in the PLAs. We further explain the use of PLAs in the industrial ecosystems. In Chapter 5, we use the PLAs to document privacy risks and the suggested controls to mitigate those risks. Practical case studies are used to evaluate the presented concepts, including the use of privacy preferences and PLAs.

Chapter 4

Model-Based Privacy Analysis

This chapter shares material with the ECMFA'17 paper "Model-Based Privacy Analysis in Industrial Ecosystems" [7] and the paper "Model-Based Privacy Analysis in Industrial Ecosystems: A Formal Foundation" [4] submitted to the SoSym Journal.

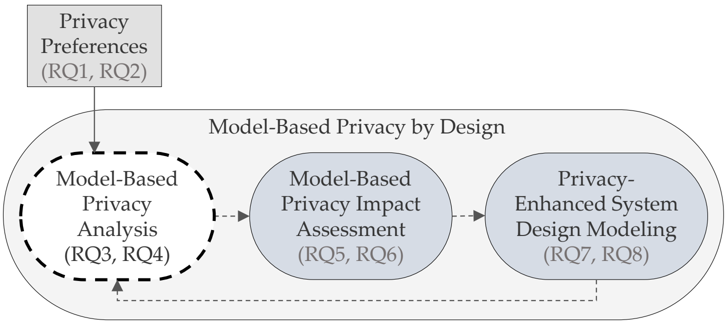

Privacy Preferences (RQ1, RQ2)
Model-Based Privacy by Design
Model-Based Privacy Analysis (RQ3, RQ4)
Model-Based Privacy Impact Assessment (RQ5, RQ6)
Privacy-Enhanced System Design Modeling (RQ7, RQ8)

Figure 4.1: The highlighted section (dashed lines) denotes how Chapter 4 contributes to the overall workflow (introduced in Section 2.3).

To integrate appropriate controls into a system design from the early phases of development—to operationalize *PbD*—the design violations of the system have to be identified. These design violations determine where precisely the controls have to be applied. The identification of such violations calls for a system analysis. In this thesis, the systems are specified with system models. System modeling enables an analysis from the onset of development. Particularly, UML [157] is used to model the systems that we analyze. In this chapter, we propose a model-based privacy analysis methodology to verify whether a set of specific privacy preferences are supported, or any enhancement in the design of the systems, which we analyze, is required. We use our definition of privacy preferences introduced in Section 3.3.

Our methodology particularly supports industrial ecosystems, where several service providers may process personal data to deliver a service. The methodology relies on the four privacy key elements, namely *purpose*, *visibility*, *granularity*, and *retention* (Section 3.2.1). We show the correctness of our model-based privacy analysis through several theorems. The privacy analysis is supported by the CARiSMA tool (Chapter 7). The methodology is applied to three practical case studies of the VisiOn project.

4.1 Introduction

Article 25 of the GDPR prescribes the *privacy by design* (*PbD*) principle [198]. *PbD* requires that service providers verify whether the required privacy levels are fulfilled in IT systems. Furthermore, it prescribes the integration of appropriate technical and organizational controls in an effective manner into IT systems from the onset of system development.

There exist a range of privacy enhancing technologies (PETs) [15, 36, 58, 73, 99, 135, 201], which provide strong privacy guarantees in different domains. However, according to Spiekermann et al. [188, 189], *PbD* is a powerful term and includes more than the process of uptaking a few PETs. Cavoukian [42], who first introduced the term *privacy by design*, defines *PbD* as the idea to integrate privacy and data protection principles in a system's design, and to recognize privacy in a service provider's management processes. Based on these considerations, *PbD* implies that the design of a system has to be analyzed with regard to privacy preferences, and where necessary, be improved to technically support privacy and data protection. In Chapter 3, relying on the four key elements of privacy (*purpose*, *visibility*, *granularity*, and *retention*), we defined privacy preferences.

System-level privacy analysis is particularly challenging in today's digital society, where industrial ecosystems play a key role. Specifically, a service provider may depend on or cooperate with other service providers to provide an IT service to a service customer. For instance, in Section 2.2, we described that *SSN* (*Social Security Number* – a piece of personal data) is required in the process of *issuing a birth certificate* in MoA (the registration office in the Municipality of Athens). We further explained that the *SSN* is not only processed by MoA, but it is transferred to other service providers, namely a tax office and a financial institute to obtain more information about the citizen, to whom the *SSN* belongs. Performing a privacy analysis in such cases requires analyzing several service providers, which process the *SSN*. To address the cases, where the system design of the relevant service providers are not entirely available, each service provider has to be analyzed individually.

In this chapter, we investigate the following research questions:

RQ3: *How can an analysis be performed on a system design in an environment where a piece of personal data is processed by several data processors?*

RQ4: *How can a system design that processes personal data be analyzed to verify whether the key elements of privacy are supported?*

To address these research questions, we make the following contributions:

- We introduce a modular methodology that analyzes the system design of the service providers separately (Section 4.3.1).
- We propose a UML privacy extension to annotate a system model with privacy relevant issues (Section 4.3.3).
- We provide a model-based methodology to analyze the system design of a service provider in regard to a set of privacy preferences. The methodology relies on the fundamental taxonomy of the four privacy key elements [22]. Based on the definition of privacy preferences proposed in the previous chapter, we show the correctness of our privacy analysis (Section 4.3.4).
- We introduce a tool-support (an extension of CARiSMA tool[1] [6]) for our model-based privacy analysis methodology (Section 7.2).
- We evaluate our methodology using three practical case studies (Section 4.4). We draw a conclusion regarding the support required by the industry partners of the project to perform a privacy analysis. This conclusion is based on a comparison between our observations and the expertise of the industry partners in system modeling, resulting from a survey, which is performed by us.

The remainder of this chapter is organized as follows. In Section 4.2, the necessary background is provided. In Section 4.3, we describe our methodology on model-based privacy analysis. In Section 4.4, we evaluate our methodology using a case study. In Section 4.5, we discuss our results. In Section 4.6, we discuss related work. Finally, in Section 4.7, we conclude.

[1] `http://carisma.umlsec.de` (accessed: 2019-06-01)

4.2 Background

In this section, we provide the necessary background for this chapter. We use UML as the *"lingua franca of software engineering"* [84] to model a system. Therefore, first, a brief background on UML is provided. Moreover, UML has to be extended to express the privacy concepts, hence, we describe the UML extension mechanism. Finally, we describe UMLsec, which is a UML security profile.

4.2.1 Unified Modeling Language (UML)

Different system models, such as informal usage for communication or learning and formal usage, are widely used in industry, and UML is the leading language in numerous software domains [193]. According to the *Object Management Group*[2] (*OMG*), the UML modeling constructs are divided into two semantic categories [157]:

- The *Structural Semantics* specifies the structure of a system model using classifiers such as classes, components, and interfaces. It provides the foundation for the behavioral semantics of UML.
- The *Behavioral Semantics* of UML is built on the structural basis to provide the required foundation for the execution of behaviors. Actions are the fundamental units of behavior in UML. They constitute activities which express the behavior of a system.

We briefly revisit the necessary technical details of class, activity, deployment and state diagrams required for our analysis. We only provide a brief description of these diagrams in this section. When we use each model for an analysis, the notation and examples are provided. Our model-based privacy analysis methodology focuses on both the structure and the behavior of the systems. The privacy analysis requires the class and activity diagrams. Deployment and state diagrams are required further to motivate and describe UMLsec (an earlier work on model-based security analysis). This section is based on the concepts provided in UML (version 2.5.1) specification [157].

[2] `https://www.omg.org/` (accessed: 2019-06-01)

4.2.1.1 Class Diagram

A *class* diagram represents the structure of a designed system with classes and their relationships—modeled as *associations*, *generalizations*, and *dependencies*. A *class* specifies a classification of objects and the features that characterize the structure and behavior of those objects. The main features of a class are its *properties* and *operation*s. A class may act as a metaclass to describe metamodels or profiles. In Section 3.4 (Figure 3.9), we presented the metamodel of a PLA using a class diagram.

Operations of a class specify the behavioral feature of the class and can be invoked on an object, given a set of values for the parameters of the operation. An operation specifies the name, type, parameters, and constraints for an invocation. The notation of an operation shown as a text string is of the form:

$$\begin{aligned}[< \mathit{visibility} >] < \mathit{name} > \text{‘(‘ } [< \mathit{parameter\text{-}list}] \text{ ‘)‘ } [\text{ ‘ : ‘ } [< \mathit{return\text{-}type} >] \\ [\text{ ‘[‘ } < \mathit{multiplicity\text{-}range} > \text{ ‘]‘ }] \; [\text{ ‘\{‘ } < \mathit{oper\text{-}property} > \\ [\text{ ‘,‘ } < \mathit{oper\text{-}property} >]^{*} \text{ ‘\}‘ }]]\end{aligned}$$

An operation has a list of parameters in the following format:

$$< \mathit{parameter\text{-}list} > ::= < \mathit{parameter} > [\text{ ‘,‘ } < \mathit{parameter} >]^{*}$$

A *parameter* is a specification of an argument used to pass information, such as a piece of personal data to be processed by an operation, into or out of an invocation of a behavioral feature (an operation). A parameter has a type and multiplicity, specifying what values may be passed and how many. Moreover, an operation has a list of properties (*oper-property*).

4.2.1.2 Activity Diagram

An *activity* diagram is a behavior indicated by a set of subordinate units, expressing the control and data flows. It is mainly used to model a business process or a workflow. An activity specifies procedural computation and may form hierarchies of activities, which invoke other activities. In an object-oriented model, an activity may be used to model the control and the data flow of the operations defined in (a) class diagram(s).

The control and data flows are modeled using *activity nodes* and *activity edges*. The activity nodes model the individual steps in an activity, thereby specifying a behavior. An activity edge is a directed connection between two nodes from the source to the target activity nodes.

There are three kinds of nodes:

- *Control Nodes* coordinate the flows between other kinds of nodes. They include initial and final nodes, decision node, merge, fork and join nodes.
- *Object Nodes* represents objects such as a piece of personal data (*activity parameter node*), or a database (*data store node*) in an activity. One specific type of an object node is a *data store node*. A data store node holds the objects persistently while its containing activity is executing. The selection and transformation behaviors can be used to get information out of a data store node as if a query is performed.
- *Executable Nodes* carry out the desired behavior of an activity and include actions as the central unit of activities. An action may call an operation of a class (*CallOperationAction*) or a behavior (*CallBehaviorAction*).
  - A *CallOperationAction* is an executable node that transmits an operation call request message to the target object and invokes the corresponding behavior. The argument values of a CallOperationAction are passed on the input parameters of the operation.
  - A *CallBehaviorAction* is an executable node that invokes a behavior directly. The argument values of a CallBehaviorAction are passed on the input parameters of the invoked behavior. A CallBehaviorAction is denoted by placing a rake-style symbol within an action notation. The contents of a CallBehaviorAction (an invoked behavior) can be shown by a flow of controls nodes, object nodes and CallOperationActions.

4.2.1.3 Deployment Diagram

A *Deployment* diagram models the relationship between physical elements of a system and the assignment of software artifacts to the physical elements. Physical system elements are represented as *physical node*s and software artifacts as *deployed artifacts*. The connection between the physical nodes are modeled using *link*s and the relationship between the artifacts are modeled using *dependencies*.

4.2.1.4 State Machines

*State machine*s are used to model the behavior of parts of a system (for instance different states of an instance of an object) or to express the interaction sequences (protocols) for parts of a system. A *state* models a situation in a state machine during which a particular condition holds. For instance, for an object, two states, namely active and inactive, may be defined. The states are connected by *transition*s. A state in a state machine is achieved by an *event* attached to a transition. An event is the specification of an occurrence that triggers a behavior.

4.2.2 UML Profile

UML metamodel can be tailored for different platforms or domains. The profiles clause describes the capabilities that allow metaclasses to be tailored. In this work, we extend the UML metamodel using UML profile's capability to allow system designers to express privacy concepts in system models. In UML, the profile's *stereotype* is used to extend UML classes.

A stereotype is a limited kind of metaclass that is used in conjunction with the metaclass that it extends. Similar to a class, a stereotype may have properties which are called *tag definitions*. The values of the properties are specified by *tagged values*. In a system model, a stereotype is denoted by writing its name in guillemets attached to an extended model element:

«*stereotype*»

The corresponding notation for the stereotypes' properties is:

«*stereotype*» {*tag = value*}

If a tag has more than a value, then the values are displayed as a comma-separated list:

«*stereotype*» {*tag = value [',' value]**}

A promising example for the UML profiles is the UMLsec profile [128]. In [157], a standard profile for UML by describing different stereotypes is provided. Different conventions may be used by profiles for naming the stereotypes. According to [157], normally, a stereotype's name starts with an upper-case. In this work, we start the name of the stereotypes with lower-case.

In the following section, we explain the UMLsec approach. It provides a foundation for the work presented in this thesis.

4.2.3 Model-Based Security Analysis Using UMLsec

UMLsec [128] provides a model-based approach to develop and analyze security critical-software, in which security requirements such as confidentiality, integrity, and availability are expressed within UML diagrams [157]. The UMLsec language is provided as a UML profile (a lightweight extension of UML using the standard UML extension mechanism) which can be imported into existing UML tools. In UMLsec, different stereotypes and tags are used to annotate UML diagrams with security properties. UMLsec provides various security checks to ensure the annotated properties. The CARiSMA tool performs the corresponding security checks. The idea of UMLsec is to provide maximal analysis power while allowing to use everyday development tools for the development process. It has been used to investigate a variety of security properties (such as [125]) in a number of applications in practice (such as [81, 111, 126, 129, 130]). Moreover, the security analysis techniques have also been applied at the code level [72] and integrated with the requirements elicitation phase [37, 182].

While the UMLsec profile is defined as a light-weight UML extension, it is also possible to define variations of it using heavyweight extensions to specify the change of semantics, as needed [178]. Thus, one can make use of an extended metamodel (analysis model) defined by a heavyweight extension. This analysis model provides the possibility of more complex analysis by extending the basic UML metamodel. To define the analysis model, the data structures for the analysis are defined and, Object Constraint Language (OCL) [156] is used to specify the constraints. Furthermore, a transformation is needed to describe how annotations (stereotypes) can be transformed into the analysis model.

As mentioned above, UMLsec provides different security checks to verify whether a security property in a system is violated, and a security mechanism is needed to restore it. In this section, we explain two security checks, namely, *secure links* and *secure dependency*. *Secure links* is used for the description and the analysis of secure data flows over connections between the artifacts in a UML deployment diagram, which describes the physical layer of a system. *Secure dependency* ensures that various dependencies between interfaces in a structure of a system model respect the security requirements of the data communicated across them.

4.2.3.1 Secure Links

As described in Section 4.2.1.3, the physical layer of a system is modeled by a deployment diagram, including physical nodes, the communications between them (modeled by links), the (software) artifacts and the dependencies between the artifacts. The *secure links* annotation enables one to ensure the security of communications in a physical layer.

In UMLsec, to perform a security check, *adversary patterns* are required. Such patterns specify the potential access paths threatened by a certain attacker. Table 4.1, represents the *default* adversary, as an example of an adversary pattern. For a given adversary of type A, the set $Threat_A(s)$ specifies which kinds of actions the adversary can apply to a node or a link marked with the stereotype s. For example, considering an unencrypted *internet* communication link, the default attacker ($Threat_{default}(internet)$) can *delete*, *read* and *insert* messages transmitted over this link.

Table 4.1: The UMLsec default adversary pattern

Stereotype s	$\boldsymbol{Threat_{default}(s)}$
«*internet*»	{delete, read, insert}
«*encrypted*»	{delete}
«*LAN*»	$\emptyset$

The stereotype «*secure links*» implies the following conditions: for each dependency annotated with stereotype $s \in \{$ «*secrecy*», «*integrity*», «*high*» $\}$ between two artifacts deployed on two nodes n, m, we have a communication link l between n and m with stereotype t such that:

- $s =$ «*high*» , implies that $threat_A(t) = \emptyset$,
- $s =$ «*secrecy*» , implies that $read \notin threat_A(t)$, and
- $s =$ «*integrity*» , implies that $insert \notin threat_A(t)$.

For instance, if a communication link between two nodes n, m are annotated with «*internet*», and the dependency between two artifacts a_1 (deployed on node n) and a_2 (deployed on node m) are annotated with «*high*», then the security constraint associated with the stereotype «*secure links*» is violated: the dependency annotated with «*high*» demands that the set of threats of an adversary is empty, however, the communication link is annotated with «*internet*», meaning that the adversary is capable of reading, deleting, or inserting messages over the link between n and m. Consequently, the security requirement of the communications is not supported.

Figure 4.2 shows an excerpt of the deployment diagram which is created for our running example (*issuing a birth certificate* case study), introduced in Section 2.2. Since the dependency between the artifacts *ApplicationForm* and *BirthCertificate* is

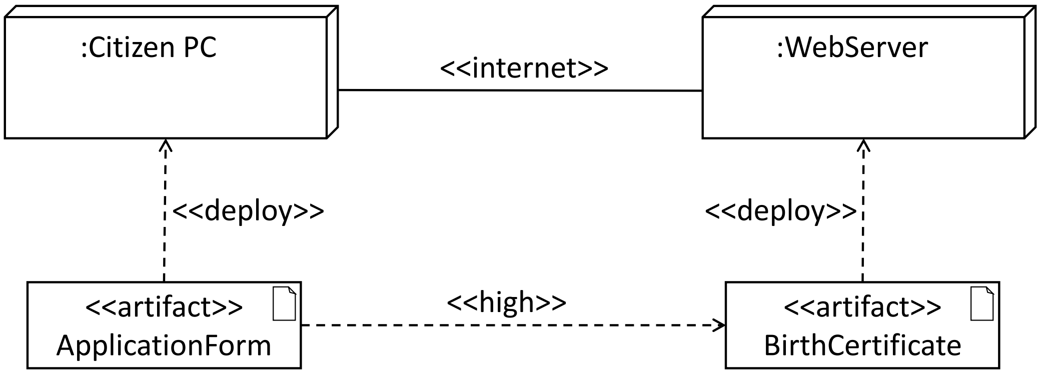


Figure 4.2: Design model excerpt (deployment diagram) annotated with «`secure links`»

annotated with «`high`», the adversary should not be able to read, delete, or insert any messages. However, the link between the nodes *Citizen* and *Webserver* is annotated with «`internet`». According to Table 4.1, a default adversary is capable of reading, deleting, or inserting a message in case of an *internet* link. Therefore, the required security level (*high*) is violated with respect to the default adversary. Since the whole diagram (deployment diagram) has to be annotated with the stereotype «`secure links`», this stereotype does not appear in Figure 4.2.

4.2.3.2 Secure Dependency

In UML, a dependency between two model elements is a relationship that denotes a model element requires other model elements for its specification or implementation. In other words, the complete semantics of the client element is either semantically or structurally dependent on the definition of the supplier element [157].

The stereotype «`secure dependency`» implies that the security requirements have to be supported by both sides of the dependency (respective classifiers) and the dependency itself. For instance, consider the design model excerpt presented in Figure 4.3, showing two classes and a dependency «`call`» between them. This figure illustrates an excerpt of the class diagram created for our running example (*issuing a birth certificate* case study). The process of requesting a birth certificate by a citizen is modeled using the *call* dependency between the two classes: *Citizen* and *CitizenRegistry*. The class *Citizen* does not implement the method *requestBirthCertificate*, therefore, it calls this method from the *CitizenRegistry* class.

The security requirements of the citizen are specified using the «`critical`» stereotype in the corresponding class. This stereotype belongs to the UMLsec

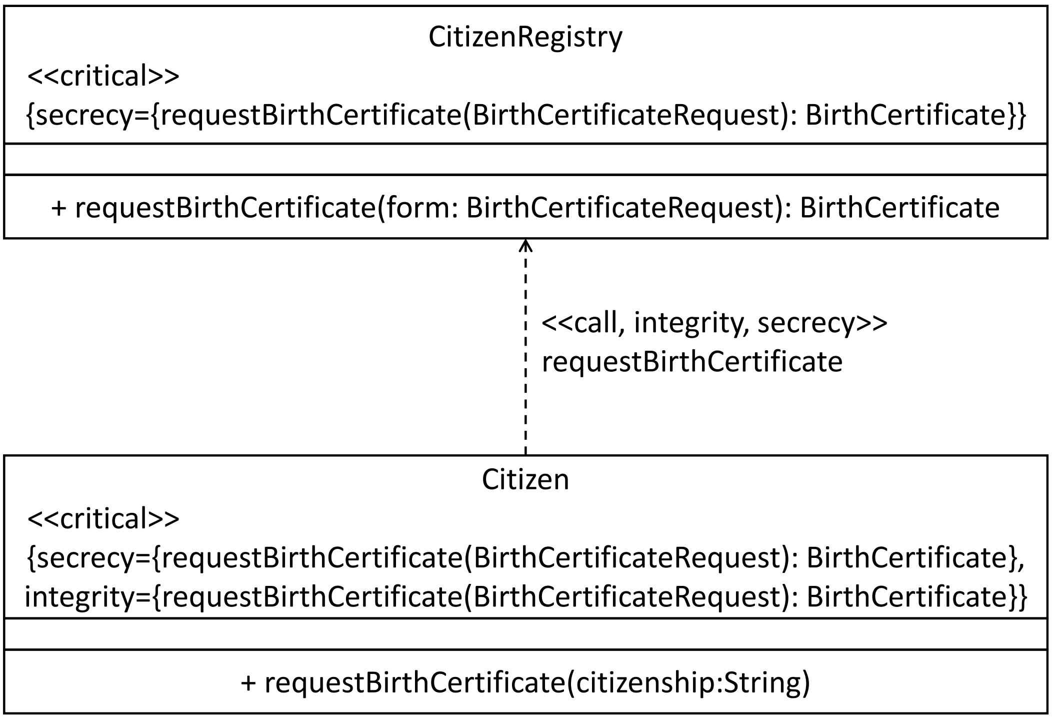

CitizenRegistry
<<critical>>
{secrecy={requestBirthCertificate(BirthCertificateRequest): BirthCertificate}}
+ requestBirthCertificate(form: BirthCertificateRequest): BirthCertificate
<<call, integrity, secrecy>>
requestBirthCertificate
Citizen
<<critical>>
{secrecy={requestBirthCertificate(BirthCertificateRequest): BirthCertificate},
integrity={requestBirthCertificate(BirthCertificateRequest): BirthCertificate}}
+ requestBirthCertificate(citizenship:String)

Figure 4.3: Design model excerpt (class diagram) annotated with «`secure dependency`»

profile and annotates the classifiers, which contain data that is critical in some way, specified by the corresponding tags, such as {*secrecy*}, {*integrity*} and {*high*} [128]. The values of these tags are the names of the attributes or methods' signatures.

In the class *Citizen*, the tags {*secrecy*} and {*integrity*} are stated for the signature *requestBirthCertificate(BirthCertificateRequest):BirthCertificate*, denoting that the citizen's data (provided to the citizen registry by requesting a birth certificate) have to be protected from unauthorized access (*secrecy*) and manipulation of a third party (*integrity*).

The stereotype «*secure dependency*» implies that both the dependency «*call*» and the *CitizenRegistry* class provide similar security requirements (*secrecy* and *integrity*). However, since the tag {*integrity*} is missing in the *CitizenRegistry* class, the security level in this example is violated.

Since the whole diagram (class diagram) has to be annotated with the stereotype «*secure dependency*», this stereotype does not appear in Figure 4.3.

4.3 Model-Based Privacy Analysis in Industrial Ecosystems

Previously in Figure 2.2, we demonstrated the overview workflow of this thesis. In this chapter, we describe our privacy analysis methodology, which is the first submethodology that has to be conducted to operationalize *PbD*. Figure 4.4 demonstrates the workflow of the privacy analysis methodology. A privacy analysis requires two main inputs:

- The annotated system model of a service provider that processes personal data. The systems to be analyzed are modeled by UML diagrams (specifically class diagrams and activity diagrams). The diagrams have to be annotated with privacy relevant issues before performing an analysis.
- The privacy preferences of personal data. In Section 3.3, we defined privacy preferences. Moreover, according to Section 3.4, a set of privacy level agreements capture the privacy preferences of personal data.

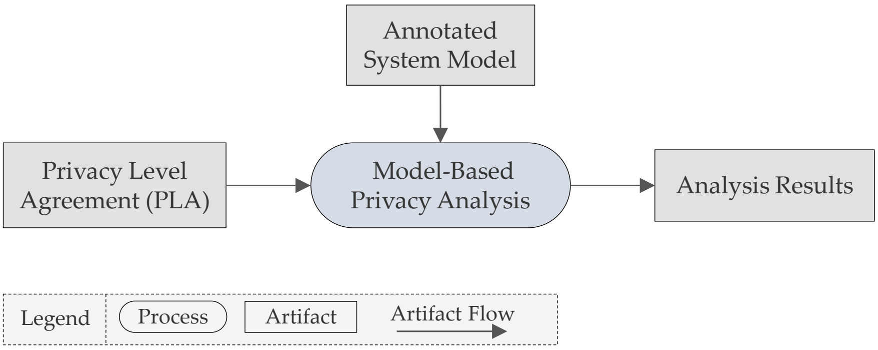


Figure 4.4: Model-based privacy analysis by exploiting PLAs

The results of a privacy analysis denote the potential privacy design violations. The analysis results may be, however, empty, indicating that the analyzed system model supports privacy preferences. The results may be subject to further evaluation, for instance, to identify privacy risks. A tool to support conducting a model based privacy analysis is provided (see Chapter 7).

4.3.1 The Modular Privacy Analysis

Figure 4.5 illustrates the service providers and the data transmissions from the *issuing a birth certificate* case study, introduced in Section 2.2. As mentioned in Section 3.4, PLAs are used to capture privacy preferences between data controllers and

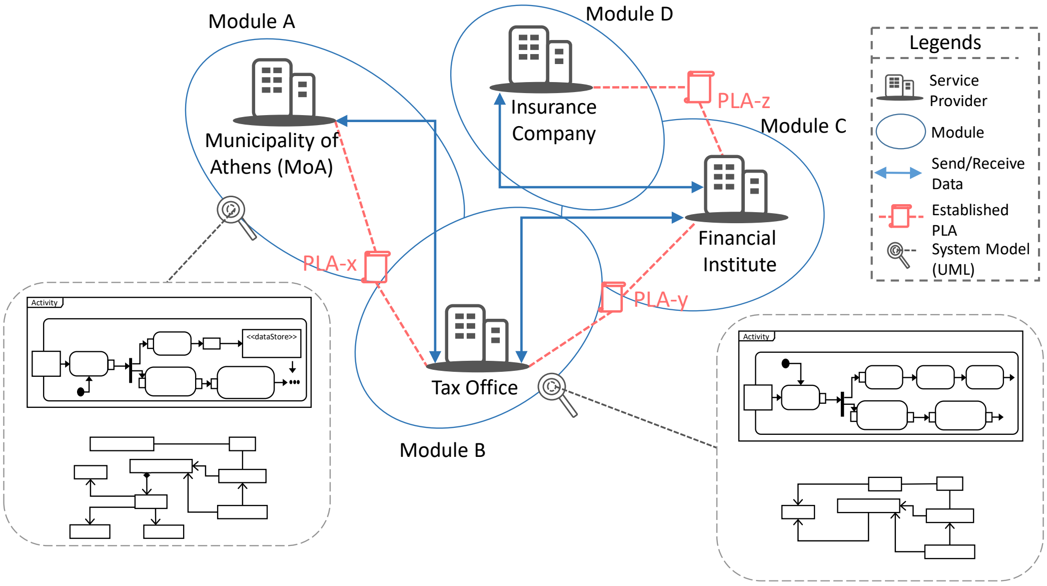


Figure 4.5: A sample illustration of an industrial ecosystem, which entails the *Municipality of Athens* (*MoA*).

data processors. In Figure 4.5, between each two service providers, a separate PLA is established, which captures the privacy preferences of the transmitted personal data. A service customer may specify that *Municipality of Athens* (MoA) is not authorized to process the *SSN* (of a citizen) for the purpose of *marketing*. This is specified in the PLA between the service customer and MoA (not demonstrated in Figure 4.5). MoA may need to verify the tax status of the service customer. Therefore it sends the personal data of a citizen (including the *SSN*) to a tax office. Between MoA and the tax office a PLA (*PLA-x*) is concluded specifying the privacy preferences of MoA. *PLA-x* has to support the privacy preferences specified for the *SSN* as well (contained in the PLA between the service customer and MoA). However, since MoA additionally may define further preferences on the use of various personal data, *PLA-x* differs from the PLA established between the service customer and MoA.

To perform a privacy analysis on such a system design, where several service providers process personal data; we need to perform a modular analysis, in which each service provider is analyzed individually. The reasons to perform modular privacy analysis are:

I PLAs are needed as input to privacy analysis. Concerning Figure 4.5, since *PLA-y* might differ from *PLA-x* and contain additional privacy preferences, the financial institute has to be analyzed individually.

II In case that the system model of one of the involved service providers is not available, a privacy analysis is still desirable. Concerning the *issuing a birth certificate* case study, if the system model of the financial institute is not available, a privacy analysis on the tax office is still possible when a modular privacy analysis is performed.

III If a new service provider is added to an industrial ecosystem or the system designs of existing ones are modified, with respect to a modular analysis, a complete analysis of the ecosystem is not mandatory.

Definition 4.3.1. ***Module:*** *In an industrial ecosystem, a module entails the structure and the behavior of* one *data processor or* one *data controller.*

Concerning Definition 4.3.1, Figure 4.5 contains four modules. MoA (*Module A*) is a processor which processes the *SSN*. When it sends the *SSN* to a tax office (*Module B*), MoA acts as a data controller that provides personal data and specifies the privacy preferences. *Module B* is a recipient of personal data and processes the *SSN*. Pursuant to Article 4 of the GDPR a recipient is a controller, or a processor, to which personal data are disclosed. *Modules B* further may send the personal data to a financial institute (*Module C*) and acts as a data controller. The financial institute and the insurance company are recipients as well.

According to Definition 4.3.1, in a modular analysis, only the structure and the behavior of one controller or one processor is analyzed. However, since the processing includes the operations or classes of other data processors (recipients), we introduce a stereotype «`recipient`» to denote the model elements of other recipients in a modular analysis. Later in this chapter, we introduce a privacy profile that is used to annotate UML diagrams with several annotations including the «`recipient`» stereotype.

4.3.2 Privacy Analysis Based on the Four Fundamental Privacy Elements

As discussed earlier, to ensure *privacy by design*, a privacy analysis is required to verify whether the privacy preferences are appropriately supported from the early phases of a system design. In this section, we highlight the importance of analyzing a system model concerning the privacy principles relating to the processing of personal data. Figure 4.6 shows an excerpt of an activity diagram, describing the process of issuing a birth certificate in the *Municipality of Athens* (*MoA*). The activity diagram processes *the social security number* (*SSN*). The *SSN* is annotated with «`sensitiveData`», expressing that the *SSN* is indeed a piece of personal data.

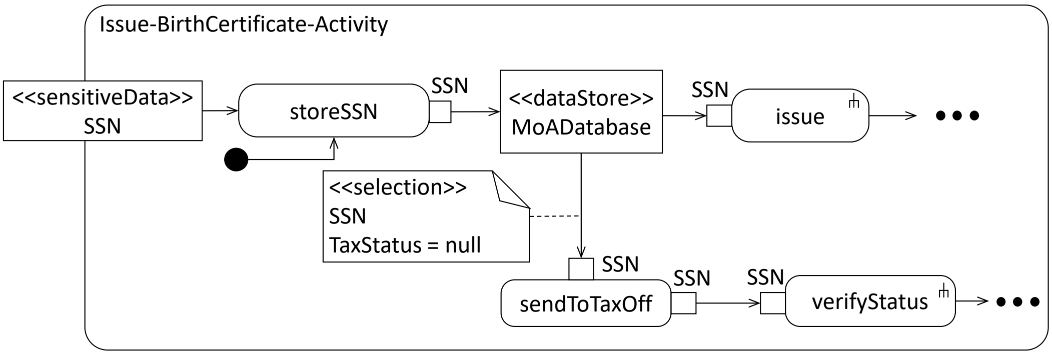


Figure 4.6: Design model excerpt (*issue birth certificate* activity), which highlights the need to perform a privacy analysis.

The sensitive data is stored (*storeSSN*) in a database (*MoADatabase*) and may be sent (*sendToTaxOff*) to a tax office to check the tax status of a citizen (*verifyStatus*).

The GDPR prescribes a set of principles on personal data processing. In 3.3, we mentioned that these principles correspond to the four key privacy elements introduced in [22] (Section 3.2.1). These key elements constitute the privacy preferences.

Concerning the activity diagram, we need to verify whether:

I The *SSN* (a piece of personal data) is only processed for the purposes that are mentioned in the privacy preferences,

II The access to the sensitive data is restricted to authorized persons,

III The granularity level is respected when sensitive data are sent to a tax office,

IV The deletion or restriction mechanisms are in place to ensure that sensitive data stored in a database, such as the *SSN*, are eventually deleted or restricted.

Considering the need to analyze these items, we propose a privacy analysis that is established upon four privacy checks. These checks correspond to the four key privacy elements.

Purpose check: Given a system model, first, the operations that process personal data and their objectives (purposes) are identified. Moreover, it has to be determined if any operation that processes personal data, belongs to other system models. For each operation that processes a piece of personal data, its objectives are compared with the purposes specified in the privacy preferences of the piece of personal data to verify whether the piece of personal data is processed only for authorized purposes.

Afterwards, for each authorized purpose for which a piece of personal data is processed, the following checks have to be performed:

- **Visibility check**: All subjects are authorized to process personal data for the specific purpose. This is verified by identifying all subjects that process personal data for the specific purpose and comparing them with the visibilities that are specified in the privacy preferences.
- **Granularity check**: Personal data is processed in regard to an authorized precision level. This is verified by identifying the required precision level (by an operation) to process a piece of data for the specific purpose and comparing the precision level with the authorized granularity levels determined by the privacy preferences of the piece of personal data.
- **Retention check**: Appropriate operations exist to restrict or delete the piece of personal data stored for the specific purpose.

Purpose check is the central part of the privacy analysis, and the rest depends on authorized purposes. In Figure 4.6, the «`sensitiveData`» stereotype is used to indicate that an object node in an activity is personal data. Annotations such as the «`sensitiveData`» stereotype enable the privacy analysis. In the following section, we introduce a complete list of stereotypes used to annotate system models.

4.3.3 UML Privacy Extension

We introduce two UML extensions, which allow one to express the privacy key elements within UML diagrams and establish a basis to perform a privacy analysis of a system model.

- First, the *privacy* profile, which is used to annotate UML models with privacy-specific information.
- Second, the *rabac* profile, which is used to generate and enforce access control policies, using the *role- and attribute based access control* model (*RABAC*, [124])

The *rabac* profile is an extension of UMLsec's *rbac* profile [128]. On top of *rbac*'s `{role}` and `{right}` tags, *rabac* allows a refined control management using an

Table 4.2: *Privacy* profile

Stereotype	Tags	UML Element	Description
«*dataPrivacy*»	data	Package	enforces privacy analysis
«*sensitiveData*»	category	NamedElement	personal data [198]
«*recipient*»	organization	NamedElement	data recipient [198]
«*granularity*»	level	Parameter	the granularity level
«*objective*»	purpose	Operation	purposes of operations

attributeFilter tag. In a nutshell, by using attributes, there is no need to increase the number of roles in a system in many cases and the problem of role explosion will be prevented. We use *rabac* to check visibility, introducing it as a separate profile, since it is not specific to privacy.

4.3.3.1 Privacy Profile

Table 4.2 lists the *privacy* profile's stereotypes together with their corresponding tags. The terms and names used in the *privacy* profile comply with the terms and the definitions of the GDPR [198] (see Section 2.1 and Appendix A).

«dataPrivacy»: A UML package is annotated with this stereotype, specifying the existence of personal data in the UML constructs of the package. Tag {*data*} specifies a set of personal data. This stereotype is used to determine whether a piece of personal data is processed within a UML package.

«sensitiveData»: A NamedElement is annotated with this stereotype together with its tag {*category*} specifying that the element is or contains sensitive data of a specific category. As we elaborated on the definition of personal data in Section 2.1, sensitive data particularly adheres to the definition of several categories of personal data:

- **commonPersonalData:** Personal data is generally defined in Article 4 the GDPR.
- **special:** Article 9 of the GDPR, Paragraph 1, refers to special categories of personal data.
- **generalIdNo:** Article 87 of the GDPR states that specific conditions for the processing of a national identification number or any other identification of general application must be determined.

Moreover, in [58], the term *privacy-relevant data* is introduced, which specifies the data that initially are not considered as personal data, however later risks for the privacy of individuals based on such data may become apparent. Therefore, we consider *privacyRelevantData* as a category of personal data, in addition to the above-mentioned three categories. This category of data includes the meta data, for instance, system log data such as timestamps that are automatically collected, and is the super group of all other personal data categories.

«recipient»: A NamedElement is annotated with this stereotype together with its tag `{organization}`, specifying that the element belongs to a controller or a processor (an organization), to which the sensitive data are disclosed.

«granularity»: A parameter is annotated with this stereotype together with its tag `{level}`, specifying the level of the data precision provided in response to a query. In Section 4.2.1.1 we showed the notation of an operation and a mentioned that an operation may have a parameter list in the following format:

$$< parameter\text{-}list > ::= < parameter > [\text{`},\text{`} \;\; < parameter >]^*$$

Each parameter in this list may be annotated with a granularity level shown using the following notation:

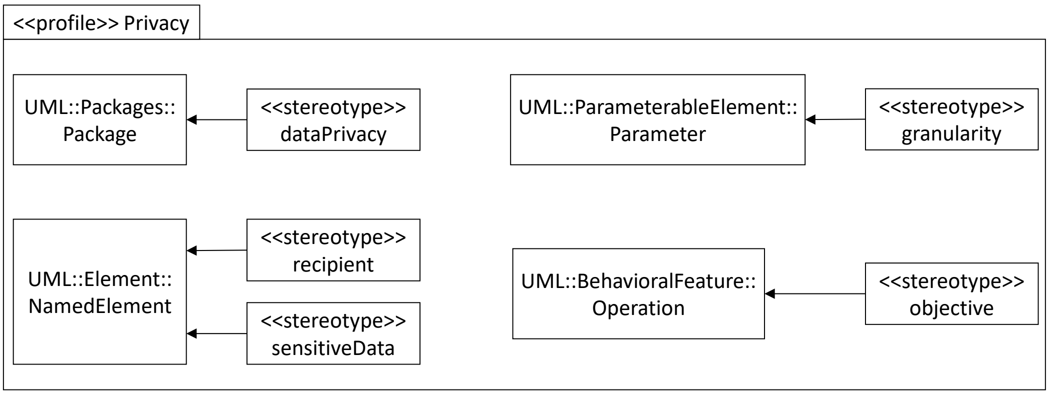
$$\text{«}\texttt{granularity}\text{»} \;::=\; \text{`}\{\text{`}\;\text{`}level\text{`}\;\text{`} = \text{`}\; value\text{-}specification\;\text{`}\}\text{`}$$

«objective»: An operation is annotated with this stereotype together with its tag `{purpose}`, specifying the purposes of the operation. Tag `{purpose}` specifies a set of processing purposes.

In Section 4.2.1.1 we showed the notation of an operation. Each operation of a class may be annotated with an objective shown using the following notation:

$$\begin{aligned}\text{«}\texttt{objective}\text{»} \;::=\; &\text{`}\{\text{`}\;\text{`}purpose\text{`}\;\text{`} = \text{`}\; value\text{-}specification \\ &[\text{`},\text{`}\; value\text{-}specification]^*\;\text{`}\}\text{`}\end{aligned}$$

Since an operation may have several objectives defined as the purposes of the operation, the values are displayed as a comma-separated list of *value- specification*s.



<<profile>> Privacy
UML::Packages:: Package
<<stereotype>> dataPrivacy
UML::ParameterableElement:: Parameter
<<stereotype>> granularity
UML::Element:: NamedElement
<<stereotype>> recipient
<<stereotype>> sensitiveData
UML::BehavioralFeature:: Operation
<<stereotype>> objective

Figure 4.7: Model of the privacy profile

In Figure 4.7, the model of the privacy profile is represented, showing the metaclasses (UML elements) to which the stereotypes are applied.

Figure 4.8 shows an excerpt of the activity provided in Figure 4.6, and a class in a class diagram of the MoA system model. The annotation «`dataPrivacy`» {`data=SSN`} specifies that a piece of personal data (*SSN*) is processed in this activity. The annotation «`sensitiveData`» {`category=generalIdNo`} specifies that *SSN* is a piece of personal data of the category *general identification number*. The annotation «`recipient`» {`organization=TaxOff`} specifies that the *verifyStatus* action belongs to the (system model of the) *tax office* organization. CallOperationAction *sendToTaxOff* induces a call to the *sendToTaxOff* operation in the *MoAIntf* interface class, expressed by the «`Trace`» dependency of the UML standard profile. The operation is annotated with «`objective`» {`purpose=[assessment, marketing]`} specifying the *assessment* and *marketing* as the purposes of the operation. The annotation «`granularity`» {`level=exact`} specifies that the exact value of the *SSN* is required for the processing.

4.3.3.2 *rabac* Profile

The list of the *rabac* profile's stereotypes together with their corresponding tags is provided in Table 4.3. *rabac* enables the verification of the visibility requirements on personal data. For each operation of a system, a set of data subjects with different roles, who are authorized to process personal data, is defined. Throughout the analysis, this information is compared to the provided privacy preferences. In what follows, the stereotypes of *rabac* together with their tags are explained. We only introduce the concepts that are relevant for the privacy analysis. More detailed information on *rabac* can be found in Section 7.2.2.3.

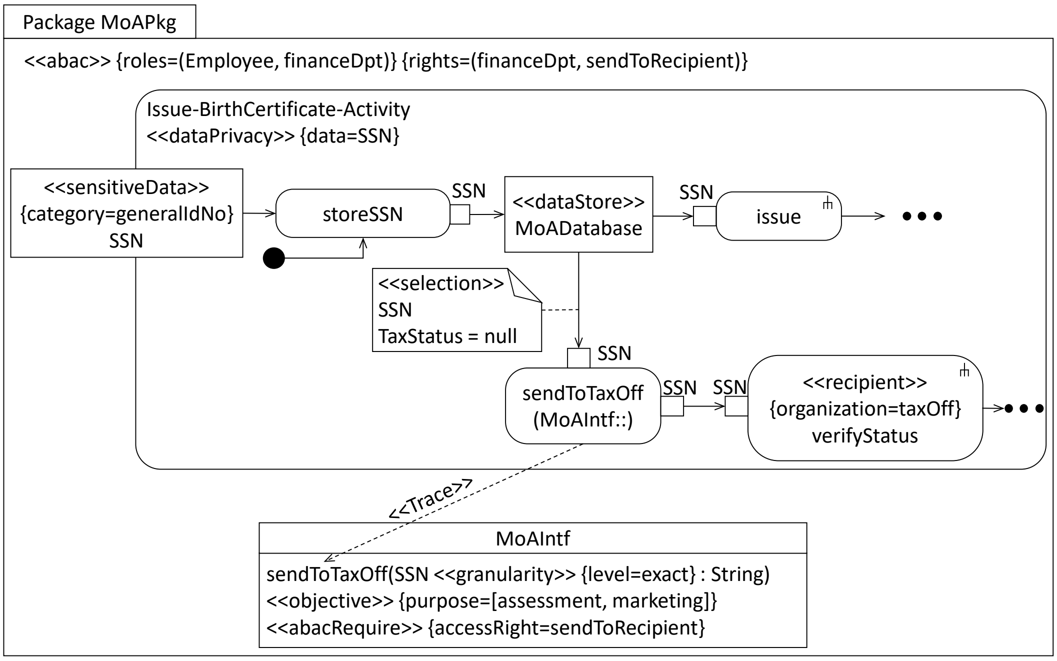

Package MoAPkg
<<abac>> {roles=(Employee, financeDpt)} {rights=(financeDpt, sendToRecipient)}
Issue-BirthCertificate-Activity
<<dataPrivacy>> {data=SSN}
<<sensitiveData>>
{category=generalIdNo}
SSN
storeSSN
SSN
<<dataStore>>
MoADatabase
SSN
issue
<<selection>>
SSN
TaxStatus = null
SSN
sendToTaxOff
(MoAIntf::)
SSN
SSN
<<recipient>>
{organization=taxOff}
verifyStatus
<<Trace>>
MoAIntf
sendToTaxOff(SSN <<granularity>> {level=exact} : String)
<<objective>> {purpose=[assessment, marketing]}
<<abacRequire>> {accessRight=sendToRecipient}

Figure 4.8: Design model excerpt annotated with the privacy and *rabac* profiles

«abac»: A package is annotated with this stereotype and its tags, namely {*roles*}, {*rights*} and {*attributeFilter*} to specify *role-attribute-based access control* is enforced in the system model. In this section, we only introduce the tags that are required for the *visibility check*. The values of {*roles*} and {*rights*} are tuples of the following form: (*dataSubject*, *associatedRole*) and (*associatedRole*, *accessRight*) respectively. The former assigns a role to a data subject, while the latter assigns a right to a role (similar to «*rbac*» [128]). Tag {*attributeFilter*} specifies a set of attributes (defined in classes) to enhance access rights. Attributes (together with roles) support a coarse-grained access model for governing which subjects may access a piece of personal data.

«abacAttribute»: An operation is annotated with this stereotype, with tag {*name*} to specify a specific attribute with a corresponding value to invoke the operation.

«abacRequire»: An operation is annotated with «*abacRequire*» with tags {*filter*} and {*accessRight*} to specify the respective attribute and the access right used to invoke the operation. Tag {*accessRight*} enables one to identify the associated roles and subjects that are used to perform the operation.

In Figure 4.8, the operation *sendToTaxOff* is annotated as follows: «*abacRequire*» {*accessRight = sendToRecipient*}. This annotation means that the relevant

Table 4.3: *rabac* profile

Stereotype	Tags	UML Element	Description
«*abac*»	roles, rights, attributeFilter	Package	enforces role-attribute-based access control
«*abacAttribute*»	name	Operation	rabac for an attribute
«*abacRequire*»	accessRight, filter	Operation	rabac for an operation

accessRight of this operation is *sendToRecipient*. Considering the stereotype «*abac*», the associated role for this *accessRight financeDpt*, who invokes the *sendToTaxOff* operation.

In the following section, we explain how the introduced profiles enable a privacy analysis.

4.3.4 The Privacy Checks

In Section 4.3.2, we introduced the abstract concepts of our privacy checks. In this section, we explain them in full detail. According to Figure 4.4, the privacy analysis requires an annotated system model with the above-mentioned profiles. Following the idea of performing a modular privacy analysis in an industrial ecosystem, the input of the privacy analysis is the system model of one module. Particularly, a privacy analysis is conducted on activity and class diagrams. In addition to a system model, the privacy preferences (Section 3.3) of the personal data are required.

The analysis examines the activity diagrams, in which a piece of personal data *pd* is processed, together with its connections to the class diagram. This is indicated by the stereotype «*dataPrivacy*» {*data=pd*}.

Purpose check

This check verifies whether a piece of personal data is processed for a set of specific authorized purposes. As explained in Section 4.2.1.2, each action in an activity diagram is either a *callOperationAction* or a *callBehaviorAction*.

- *(**Case I**) The action is a callOperationAction:* The action that processes *pd*, refers to an operation in a class of the class diagram that represents the structure of the system model. In a system model, this is denoted by the «*Trace*» dependency between the action and the operation. Using the annotation «`objective`» (privacy profile), for each operation, a set of processing purposes is defined.

According to Lemma 3.3.1, given a set of *base* privacy preferences of a piece of personal data pd, in form of $PrP_{pd} = \{b_1, b_2, ..., b_n\}$, the set of all authorized purposes specified by the base privacy preferences may be indicated by the set $\mathcal{P}_{pd}$:

$$\mathcal{P}_{pd} = \bigcup_{i=1}^{n} l.u.b.(PL)/p_{b_i}$$

For each objective o of an operation, we verify:

$$o \in \bigcup_{i=1}^{n} l.u.b.(PL)/p_{b_i}$$

The check identifies all the objectives of the operation, which corresponds to an action in the activity diagram. Afterwards, for each objective o, it is verified whether o is a member of the set of authorized processing purposes ($\mathcal{P}_{pd}$), determined by the privacy preferences of the pd.

- *(**Case II**) The action is a callBehaviorAction:* In this case, the activity that is invoked by the *callBehaviorAction* is analyzed. Such an invocation continues until all actions that process personal data are *callOperationAction*s.

Discovering an objective o (of an operation that processes a piece of personal data pd) which is not a member of the set of authorized purposes ($o \notin \bigcup_{i=1}^{n} l.u.b.(PL)/p_{b_i}$), is a privacy design violation. Such a violation indicates that a piece of personal data is processed for (an) unauthorized purpose(s). This particularly violates Article 5, Paragraph 1 (b) and (c) of the GDPR, where it is prescribed that:

> "(b) Personal data shall be collected for specified, explicit and legitimate purposes and not further processed in a manner that is incompatible with those purposes [...]."

> "(c) Personal data shall be adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed."

Theorem 4.3.1 (Correctness of purpose check)**.** *Given a system model including activity diagram A and class diagram C annotated with the privacy and rabac profiles and a set of base privacy preferences $PrP_{pd} = \{b_1, b_2, ..., b_n\}$ for a piece of personal data pd, the purpose check identifies all privacy design violations resulting from processing pd for specific unauthorized purposes within the activity A.*

Proof. Let the following be given:

- An activity A as a tuple $\langle Name, Nodes, Edges \rangle$[3], where:
  - $Nodes = \langle ON, CN, EN \rangle$ where ON is a set of object nodes, and CN is a set of control nodes,
  - and EN is a set of executable nodes (actions), so that $EN = \langle COA, CBA \rangle$ with COA being a set of callOperationActions, and CBA being a set of callBehaviorActions,
- A class diagram $C = \langle Name, Classes, Associations \rangle$, where $Classes = \langle OP, ATT \rangle$ with OP being a set of operations, and ATT being a set of attributes,
- Each $coa \in COA$ is traced to an $op \in OP$,
- Each $cba \in CBA$ invokes a behavior, which is modeled with an activity including a set of callOperationActions,
- A piece of personal data pd is modeled as an object node in activity A, and is annotated (using *privacy* profile) with «`sensitiveData`»,

Lemma 3.3.1 specifies the set of purposes $\mathcal{P}_{pd}$ for which pd is authorized to be processed:

$$\mathcal{P}_{pd} = \bigcup_{i=1}^{n} l.u.b.(PL)/p_{b_i}$$

The behavior of A is modeled by the set EN, where (EN) eventually contains only a set of callOperationActions (COA), and each $coa \in COA$ is traced to an operation $op \in OP$, for which (using *privacy* profile) a set of objectives (processing purposes) $O_{op} = \{o_1, ..., o_i\}$ is indicated, thereby, specifying the processing purpose(s) of coa. For each $o \in O_{op}$, the *purpose check* verifies whether $o \in \mathcal{P}_{pd}$. Hence, the *purpose check* analyzes the activity A to verify whether pd is processed for only a set of specific authorized purposes, and if there is an $o \notin \mathcal{P}_{pd}$, this is identified as a privacy design violation. □

[3] Based on the semantics of UML 2.0 Activities introduced by Störrle in [192].

After checking purpose, for each authorized objective o, which is a member of the set of authorized processing purposes determined by the privacy preferences of pd, the *visibility*, *granularity*, and *retention* checks have to be performed.

Visibility check

According to Lemma 3.3.2, given a set of *base* privacy preferences of a piece of personal data pd, in form of $PrP_{pd} = \{b_1, b_2, ..., b_n\}$, for each purpose p_{b_i} included in the element b_i, the set of authorized subjects is denoted by a join-semilattice $(V_{b_i}, \vee)$.

The operation that has the objective o, is annotated with an access right $right$ («`abacRequire`» `{accessRight = right}`). Using the stereotype «`abac`», the set of associated roles $R = \{r_1, r_2, ..., r_n\}$, to which the access right $right$ is assigned, may be identified. The *visibility check* verifies whether: $R \subseteq (V_{b_i}, \vee)$.

Notably, objective o is verified as an authorized purpose to process pd, and $(V_{b_i}, \vee)$ specifies the set of subjects who are authorized to process pd for purpose o (according to the privacy preferences).

The check identifies all the roles who process pd for the objective o. Afterwards, it verifies whether this set of roles (R) is a subset of the set of subjects who are authorized to process pd for purpose (objective) o.

Discovering a role r ($r \in R$), which is not an element of the set of authorized subjects, is a privacy design violation ($r \notin (V_{b_i}, \vee)$). Such a violation indicates that a piece of personal data is processed for a specific purpose by (an) unauthorized subject(s). This particularly violates Article 5, Paragraph 1 (e) of the GDPR, where it is prescribed that:

> "Personal data shall be kept in a form which permits the identification of data subjects for no longer than is necessary for the purposes for which the personal data are processed [...]."

Theorem 4.3.2 (Correctness of visibility check)**.** *Given a system model including activity diagram A and class diagram C annotated with the privacy and rabac profiles, and a set of base privacy preferences* $PrP_{pd} = \{b_1, b_2, ..., b_n\}$ *for a piece of personal data pd, the visibility check identifies all privacy design violations resulting from processing pd for a specific purpose* p_{b_i} *by specific unauthorized subjects within the activity A.*

Proof. According to Lemma 3.3.2, the set of subjects who are authorized to process

pd for purpose p_{b_i} can be specified by the join-semilattice $(V_{b_i}, \vee)$.

Following Theorem 4.3.1, a piece of personal data pd is modeled as an object node in activity A, and is annotated with «`sensitiveData`». Moreover, the behavior of A is modeled by a set of executable nodes EN, where (EN) eventually contains only a set of callOperationActions (COA), and each $coa \in COA$ is traced to an operation $op \in OP$. For each operation op with respect to «`abacRequire`» and «`abac`» annotations, a set of associated roles $R_{op} = \{r_1, r_2, ..., r_n\}$ is identified, who may invoke (have access to) operation op. R_{op} specifies a set of subjects.

The *visibility check*, given purpose p_{b_i}, verifies whether $R_{op} \subseteq (V_{b_i}, \vee)$. Hence, given purpose p_{b_i}, the *visibility check* analyzes activity A to verify whether pd is processed for purpose p_{b_i} by only a set of specific authorized subjects, and if there is a role (subject) in R_{op} which is not an element of the $(V_{b_i}, \vee)$, this is identified as a privacy design violation. □

Granularity check

According to Lemma 3.3.3, given a set of *base* privacy preferences of a piece of personal data pd in form of $PrP_{pd} = \{b_1, b_2, ..., b_n\}$, for each purpose p_{b_i} included in the element b_i, the interval sublattice $l.u.b.(GL)/g_{b_i}$ specifies the set of authorized granularity levels.

The stereotype «`granularity`» `{level=g}` specifies the granularity level g, which is required by the operation, whose *objective* is o, to process pd. The stereotype annotates the parameter pd of the operation.

The check verifies whether $g \in l.u.b.(GL)/g_{b_i}$. It is verified whether the granularity level g, which is required by an operation to process pd, is a member of the set of authorized granularity levels $(l.u.b.(GL)/g_{b_i})$ to process pd for objective o (concerning the privacy preferences of pd).

Discovering a granularity level g, which is not an element of the set of authorized granularity levels, is a privacy design violation ($g \notin l.u.b.(GL)/g_{b_i}$). Such a violation indicates that a piece of personal data is processed for a specific purpose with an unauthorized level of precision. This particularly violates Article 5, Paragraph 1 (d) of the GDPR, where it is prescribed that:

> "Personal data shall be accurate and, where necessary, kept up to date [...]."

Theorem 4.3.3 (Correctness of granularity check)**.** *Given a system model including activity diagram A and class diagram C annotated with the privacy and rabac profiles, and a set of base privacy preferences $PrP_{pd} = \{b_1, b_2, ..., b_n\}$ for a piece of personal data pd, the granularity check identifies all privacy design violations resulting from processing pd for a specific purpose p_{b_i} with a specific unauthorized precision level.*

Proof. According to Lemma 3.3.3, the set of granularity levels, with which pd is authorized to be processed for purpose p_{b_i}, may be denoted by an interval sublattice $l.u.b.(GL)/g_{b_i}$.

Following Theorem 4.3.1, a piece of personal data pd is modeled as an object node in activity A, and is annotated with «`sensitiveData`». Moreover, the behavior of A is modeled by a set of executable nodes EN, where (EN) eventually contains only a set of callOperationActions (COA), and each $coa \in COA$ is traced to an operation $op \in OP$. Each operation op has a list of parameters $<$ *parameter-list* $>$ ::= $<$ *parameter* $>$ [‘,‘ $<$ *parameter* $>$]*, where to each parameter pa (using *privacy* profile) a precision level g_{pa} is assigned. The parameter pa can be traced to an object node in activity A annotated with «`sensitiveData`» (a piece of personal data pd). In this case, given purpose p_{b_i}, *granularity check* verifies whether $g_{pa} \in l.u.b.(GL)/g_{b_i}$. Hence, given purpose p_{b_i}, the *granularity check* analyzes activity A to verify whether pd is processed for purpose p_{b_i} with a specific authorized precision level and if there is a $g_{pa} \notin l.u.b.(GL)/g_{b_i}$, this is identified as a privacy design violation. □

Retention check

According to Article 5, paragraph 1 (e) of the GDPR:

> "Personal data shall be kept in a form which permits the identification of data subjects for no longer than is necessary for the purposes for which the personal data are processed [...]."

A piece of personal data has to be stored for no longer than is necessary and after processing a piece of personal data for a specific purpose, the piece of personal data has to be deleted or restricted. During a system design and by means of a system model, it is not possible to certainly verify whether a piece of personal data will be eventually deleted or restricted at some point after data processing. Such a verification requires a runtime analysis. Therefore, in our privacy analysis, we basically verify whether an appropriate mechanism exists in a system model to delete or restrict a piece of personal data.

The *retention check* verifies whenever a piece of personal data is stored in a database, an action exists to eventually restrict access to or delete this data. In an activity diagram, a database is denoted by a *data store* node (Section 4.2.1.2). If in an activity diagram, a piece of personal data (an object annotated with «`sensitiveData`») is stored in a node annotated with «`dataStore`», a selection on the «`dataStore`» has to retrieve the piece of personal data and subsequently an action with *restrict* or *delete* objective has to restrict or delete the piece of personal data. If such a selection and subsequently, such an action does not exist to restrict or delete a piece of stored personal data, a privacy design violation occurs. Such a violation indicates that since no appropriate mechanism is available to delete or restrict personal data, a piece of personal data may unnecessarily be kept or stored in a system.

Theorem 4.3.4 (Correctness of retention check)**.** *Given a system model including activity diagram A and class diagram C annotated with the privacy and rabac profiles, and a set of base privacy preferences* $PrP_{pd} = \{b_1, b_2, ..., b_n\}$ *for a piece of personal data pd, the retention check identifies all privacy design violations resulting from processing pd for a specific purpose* p_{b_i} *where no appropriate mechanism exists in activity diagram A to delete or restrict pd.*

Proof. Let the following be given:

- A piece of personal data pd is modeled as an object node in activity A, and is annotated with «`sensitiveData`» (see Theorem 4.3.1).
- The behavior of A is modeled by a set of executable nodes EN, where (EN) eventually contains only a set of callOperationActions (COA), and each $coa \in COA$ is traced to an operation $op \in OP$ (see Theorem 4.3.1).
- A database is modeled as a data store node in an activity diagram, denoted by an object node annotated with «`dataStore`».

The *retention check*, for a piece of personal data pd which is stored in a data store node ds in an activity A, verifies whether a selection behavior exists that offer pd to a callOperationAction $coa \in COA$, where coa is traced to an operation $op \in OP$, for which (using *privacy* profile) either *delete* or *restrict* objective is specified. Hence, given a processing purpose p_{b_i}, the *retention check* analyzes activity A to verify whether pd is processed for purpose p_{b_i}, then an appropriate mechanism exists that delete or restrict pd, and if such a mechanism does not exists, this is identified as a privacy design violation. □

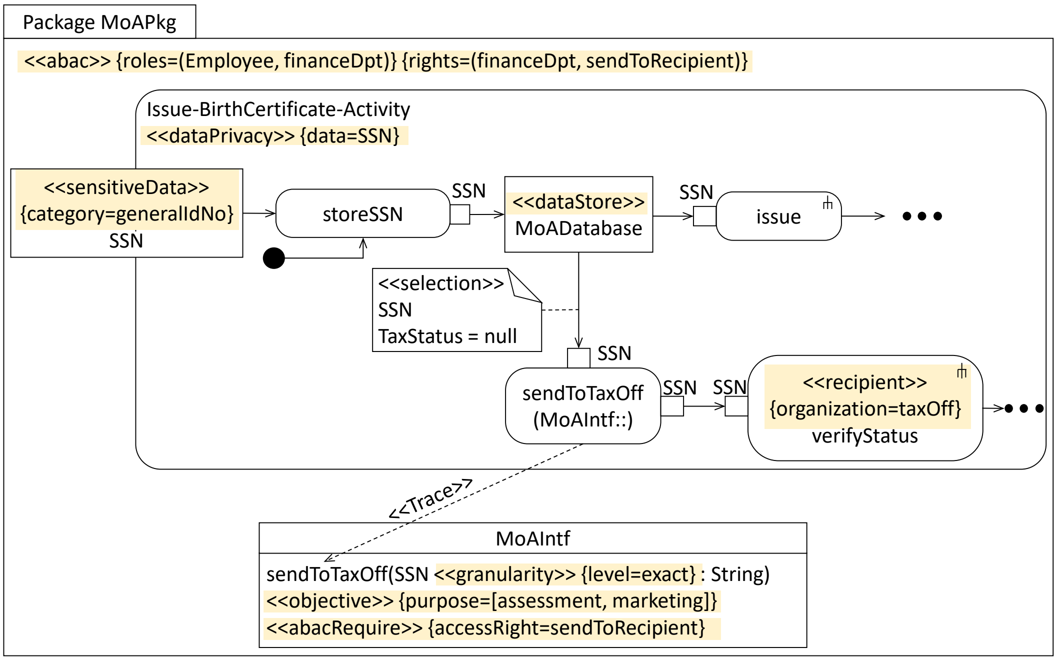

Package MoAPkg
<<abac>> {roles=(Employee, financeDpt)} {rights=(financeDpt, sendToRecipient)}
Issue-BirthCertificate-Activity
<<dataPrivacy>> {data=SSN}
<<sensitiveData>>
{category=generalIdNo}
SSN
storeSSN
SSN
<<dataStore>>
MoADatabase
SSN
issue
<<selection>>
SSN
TaxStatus = null
SSN
sendToTaxOff
(MoAIntf::)
SSN
SSN
<<recipient>>
{organization=taxOff}
verifyStatus
<<Trace>>
MoAIntf
sendToTaxOff(SSN <<granularity>> {level=exact} : String)
<<objective>> {purpose=[assessment, marketing]}
<<abacRequire>> {accessRight=sendToRecipient}

Figure 4.9: Design model excerpt with highlighted annotations for a better understanding of the privacy checks

Theorems 4.3.1-4.3.4 ensure that the proposed privacy checks can reliably detect privacy design violations related to the fundamental elements of privacy, assuming that the input system and the privacy preferences are specified correctly. We discuss related limitations in Section 4.5.

4.3.5 Applying the Privacy Checks: Example

Figure 4.9 shows the annotated activity diagram that we previously introduced. We apply the privacy checks that we introduced in the previous section to this activity diagram. The relevant annotation are highlighted for a better understanding. It has to be noted that this figure is an excerpt of the activity diagram that models the behavior of *issuing a birth certificate* (Section 2.2). We focus on the privacy analysis of the *sendToTaxOff* action, which accepts the *social security number* (*SSN*) as its input.

In Section 3.3, we introduced a sample set of privacy preferences for SSN. The same privacy preferences are used to analyze the activity diagram:

$$PrP_{SSN} = \{(assessment \mapsto (\{financeDept, saleDept\}, partial, 1year))\}$$

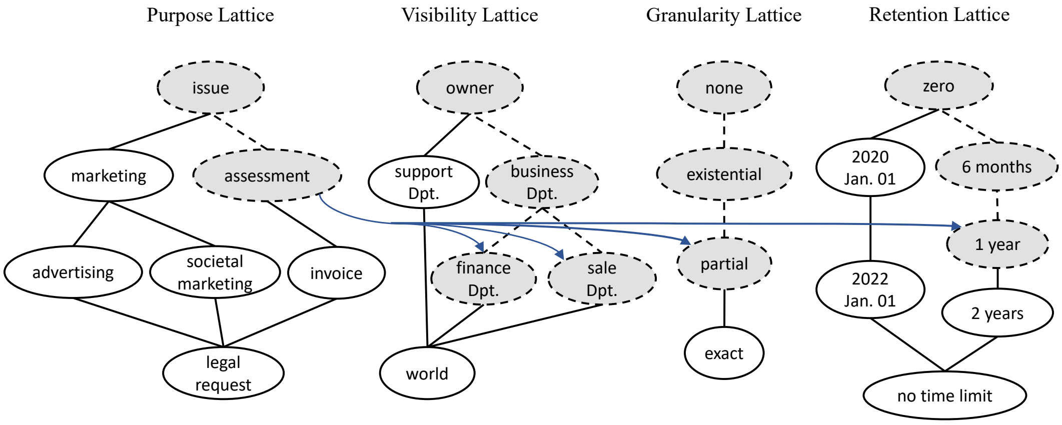


Figure 4.10: The privacy preferences of *SSN* (see Section 3.3). The dashed lines indicate the privacy preferences (authorized purposes, subjects, granularity levels, and retention conditions).

Figure 4.10 shows four lattices, including the privacy preferences of *SSN* denoted by dashed lines.

The *sendToTaxOff* action is a *callOperationAction*, for which an operation in a class of the corresponding class diagram of the system exists—this is shown by the dependency annotated with «`Trace`».

Purpose check: The *sendToTaxOff* operation processes *SSN* for two purposes, namely *marketing* and *assessment* («`objective`» `{purpose=[assessment, marketing]}`). According to the privacy preferences (purpose lattice), *SSN* is only authorized to be processed for *assessment* and the purposes that are subsumed (see Definition 3.3.2) by this purpose. Processing *SSN* for *marketing* is unauthorized (a privacy design violation). Performing the privacy analysis results in the following privacy design violation (concerning the *sendToTaxOff* action):

> "*SSN* is processed for the unauthorized purpose *marketing*."

For the authorized objective *assessment*:

- *Visibility check:* The required access right of the *sendToTaxOff* operation to process *SSN* is *sendToRecipient* («`abacRequire`» `{accessRight = sendToRecipient}`). In regard to the «`abac`» stereotype, this right is assigned to the finance department (*financeDpt*). This indicates that *financeDpt* has access to *SSN*. Concerning the visibility lattice (Figure 4.10), *financeDpt* is an authorized subject to process *SSN* for the *assessment* purpose.

- *Granularity check:* The *sendToTaxOff* operation requires *SSN* with the precision level *exact* for its processing. Concerning the granularity lattice (Figure 4.10), *SSN* is only authorized to be processed with *partial* or *existential* precisions. Therefore, the privacy analysis results in the following privacy design violation:

  > "*SSN* is processed with the unauthorized precision level *exact* for the authorized *assessment* purpose."

- *Retention check:* Pursuant to Article 5, paragraph 1 (e) of the GDPR, *SSN* has to be kept no longer than is necessary for the *assessment* purpose. Concerning the retention lattice (Figure 4.10), *SSN* has to be removed or restricted after *1 year* that it has been processed for the *assessment* purpose. As previously mentioned, we may only verify the existence of an appropriate mechanism to remove or restrict *SSN* (if in an activity, it is stored in a *dataStore* node).

  Concerning the activity diagram, *SSN* is stored in *MoADatabase* (a *dataStore* node), however, no selection and subsequently no appropriate operation with *restrict* or *delete* objective are available. Hence, the privacy analysis results in the following privacy design violation:

  > "*SSN* may unnecessarily be kept in the system after it has been processed for the *assessment* purpose."

In summary, the analysis finishes with three results, the first one related to an unauthorized processing of the SSN for a specific purpose, the second one related to a prohibited granularity level and the third one related to the absence of a required retention mechanism." The privacy design violations that are identified by performing a privacy analysis are later (in Chapter 5) used to perform a *privacy impact assessment*.

4.4 Case Studies and a Survey

In this section, we explain the application of our model-based privacy analysis methodology to three practical case studies of the VisiOn project. Furthermore, we describe a survey that we performed in the same project to assess the expertise of the target user group of our methodology in system modeling. The survey does not aim to evaluate the proposed model-based privacy analysis. It assists us to investigate the support required by the users to model their systems and to carry out the proposed methodology.

4.4.1 Case Studies

In Section 2.2, we introduced a running example based on one of the case studies of the *VisiOn* project. As previously mentioned, the VisiOn privacy Platform (VPP) evaluates the privacy levels of a *public administration* (*PA*) system. It further generates privacy level agreements on the use of personal data between citizens and PAs. VPP includes several model-based tools to perform privacy and security analyses of a PA system. VPP additionally includes other tools, for instance, to provide a graphical representation, gathering data and a database

In the course of the VisiOn project, the system models of three public administrations (which were our industry partners in the project) were used as input to the modeling-tools. The UML system models were used to evaluate the concepts proposed in this thesis. Particularly, we applied our privacy analysis methodology to these system models.

More details on the VPP and its architecture is presented in Section 7.2. Our model-based privacy analysis is supported and implemented by the CARiSMA tool. CARiSMA is a privacy and security analysis tool [6], and it is integrated into the VPP. We elaborate more on CARiSMA and its integration into VPP in Section 7.2.

The three *public administration*s:

- *DAEM SA, the IT company of the Municipality of Athens (MoA)*[4]: MoA is introduced in Section 2.2. MoA provides different online administrative services to the citizens of Athens. It cooperates with several organizations, public administrations, and enterprises.

- *Bambino Gesù Hospital (OPBG) in Rome*[5]: OPBG is a children's hospital in Rome. OPBG uses the personal data of the patients to provide different services to the patients and medical staff. VPP particularly analyzes the data transfer between OPBG and the *Hospital University of Niño Jesús (HUNJ)*[6].

[4] `http://www.daem.gr/` (accessed: 2019-06-01)

[5] `http://www.ospedalebambinogesu.it/en/home` (accessed: 2019-06-01)

[6] `http://www.madrid.org/cs/Satellite?pagename=HospitalNinoJesus/Page/HNIJ_home` (accessed: 2019-06-01)

Table 4.4: Information on the three case studies

Name	Domain	UML elements	Annotations
MoA	Urban administration	141	33
OPBG	Public health	167	31
MISE	Economic development	309	57

- *Ministry of Economic Development (MISE), in Rome*[7]: MISE performs different tasks such as verifying solvency or generating tax declarations by processing personal data. It cooperates with several organizations, and PAs.

Concerning the UML-based system models, particularly, class diagrams, activity diagrams, deployment diagrams, and state machines are used to model the PA systems. The UML diagrams are further annotated with the UML extensions proposed in this chapter (Section 4.3.3) and the UMLsec profile [128] (explained in Section 4.2.3). Table 4.4 provides information on the size of the system models and the number of annotations. The PAs modeled their system and afterwards, annotated the system models. We supported the PAs with the two tasks: modeling and annotating. Later in this section, we elaborate on our support. The three system models can be found online[8].

Results of applying the proposed privacy analysis to the three case studies include that our approach can be successfully applied to three software systems in an industrial ecosystem with complex structures and behaviors. Concerning the research questions investigated in this chapter, the results include the following:

RQ3: How can an analysis be performed on a system design in an environment where a piece of personal data is processed by several data processors? We defined the term "module" in industrial ecosystems concerning the system's design of IT services and we introduced modular privacy analysis in such ecosystems, in the sense that data processors that cooperate with each other to process personal data are analyzed separately. A modular analysis allows verifying system models in an environment where various privacy preferences for a piece of personal data is specified. In a real-world setting, most likely, the data processors have to consider distinct regulations and preferences to process personal data. Furthermore, in case that a new service provider is added to the environment where personal data are processed, or a service provider is modified, repeating a complete analysis is not necessary. Through a modular analysis, only the added or modified system has to be analyzed. This, in fact, improves the efficiency of verifying system model with respect to privacy preferences.

[7] `https://www.mise.gov.it/index.php/en/` (accessed: 2019-06-01)

[8] `https://cloud.uni-koblenz-landau.de/s/ocRXY9nJqDWzgpA` (accessed: 2019-06-01)

RQ4: How can a system design that processes personal data be analyzed to verify whether the key elements of privacy are supported? We introduced a UML privacy extension (the privacy profile and *rabac* profile) to enable four privacy checks to analyze a system model based on the four key privacy elements. We explained our proposed model-based privacy analysis to identify privacy design violations in the *issuing a birth certificate* activity. We further showed the correctness of our privacy checks with four theorems.

4.4.2 Investigating the Required Support to Carry out the Proposed Methodology

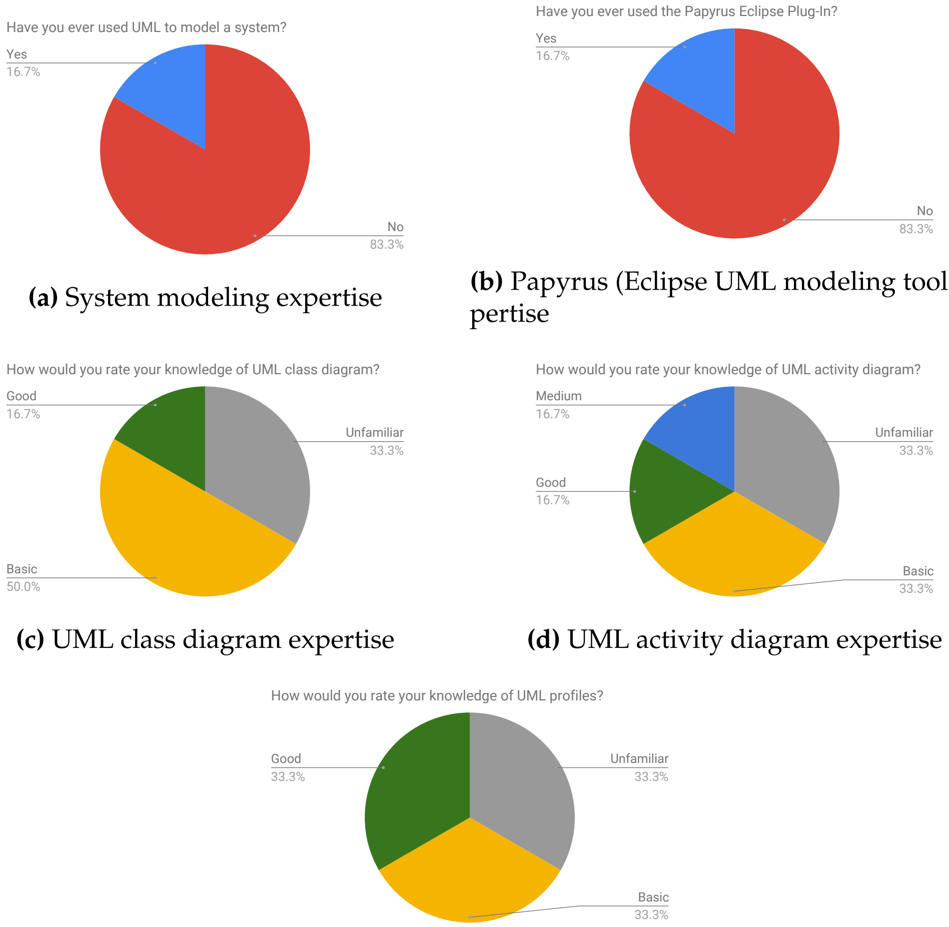
To investigate the required support to create an appropriate system model —which acts as an input to our proposed privacy analysis methodology—in an industrial ecosystem: **(I)** We performed a survey with the aim to investigate the expertise of the industry partners in system modeling. **(II)** We validated the annotated system models provided by the industry partners. **(III)** We draw a conclusion regarding the required support to carry out our proposed methodology by comparing our observations resulting from the system models and the results of the survey. **(IV)** Eventually, concerning the results of the case studies and the reports provided by our industry partners, we discuss our proposed model-based privacy analysis regarding its applicability.

4.4.2.1 A Survey on System Modeling within the VisiOn Project

The survey was constructed by close cooperation of the technical partners participated in the VisiOn project and the VisiOn management team. As previously mentioned, the VisiOn Privacy Platform (VPP) includes several modeling tools. The tool owners contributed to the construction of the questionnaire (survey) by providing their tool relevant questions. We (as the owner of the CARiSMA tool) were interested in the questions related to UML and Eclipse[9].

The questionnaire (from our perspective) has two main parts, the UML related questions, and the Eclipse-related questions. The UML part includes the questions on the required knowledge of several UML diagrams. The questions may be answered by *Unfamiliar*, *Basic*, *Medium*, *Good*, and *Advanced*. The Eclipse part includes the questions on the required knowledge of using Eclipse in general and

[9] The CARiSMA tool is based on the Eclipse IDE (`https://www.eclipse.org/`) (accessed: 2019-06-01)



(a) System modeling expertise

(b) Papyrus (Eclipse UML modeling tool) expertise

(c) UML class diagram expertise

(d) UML activity diagram expertise

(e) UML profile expertise

Figure 4.11: An excerpt of the survey results showing the expertise of our industry partners in system modeling

the Papyrus Eclipse Plugin[10]. The questions may be answered by *yes* or *no*.

Since the main aim of the survey was to gather feedback on the knowledge of the Public Administration (PA) staff, who later modeled the PA systems. Overall, six employees of the PAs were asked.

The survey is executed by the VisiOn project management team, in October and November 2016. *SurveyMonkey*[11], an online cloud-base survey development software, was chosen to perform the survey. The participants had generally at least a

[10] Papyrus is a UML open-source tool based on Eclipse, `https://www.eclipse.org/papyrus/` (accessed: 2019-06-01)

[11] `https://www.surveymonkey.com/` (accessed: 2019-06-01)

basic understanding of computer science. An excerpt from the results is shown in Figures 4.11[12]. The shown percentages in each pie chart accumulate the answers of the participants to the related question. We found that the participants rarely have used UML to model a system (16.7%). Moreover, their expertise on Papyrus (which is needed to model and annotate a system) is low (only 16.7% of the participants have already used Papyrus). The majority of the participants are unfamiliar with UML diagrams (particularly, class and activity diagrams), or have basic knowledge of them. Furthermore, only 33.3% of the participants rated their skill on UML profile with "good" level. In regard to the Eclipse knowledge, they generally know Eclipse (except one participant), however they have little or no knowledge of the Eclipse views, perspectives (not shown in Figure 4.11).

Threats To Validity The population studied in the survey is limited and may be not representative for all public administrations. Moreover, only the public administrations that participated in the project were considered in the survey. In the course of our current EU project[13], we plan to perform a more rigorous statistical analysis in cooperation with our industry partners. Furthermore, the expertise of the participants were assessed by a subjective questionnaire rather than an objective assessment. Finally, the assessment of the current usefulness of our model-based privacy analysis methodology is mostly based on the studies that are planned and executed by our industry partners. In the future, we plan to perform a survey, which involves the assessment of the usefulness of the proposed methodology.

4.4.2.2 Our Observation and Conclusion

The survey results denoted that generally, the majority of participants had no or basic knowledge of system modeling. Our direct communication, through the physical and virtual project meetings and Emails, with the associated PA staff confirmed the results of the survey.

Hence, to train the PA staff, we provided a set of screencasts and manuals (Eclipse Help Content, training activities manual [44]), a webinar[14] and a workshop[15].

Following our training, our industry partners (PA staff) provided us three system models, each modeling a PA system. Table 4.4 showed some relevant details on the three system models. The annotation of the system models is also performed by the

[12] The complete results of the survey can be found online: https://cloud.uni-koblenz-landau.de/s/ocRXY9nJqDWzgpA (accessed: 2019-06-01)

[13] *Qu4lity* project (EU's Horizon 2020 Program, No. 825030), https://cordis.europa.eu/project/rcn/220162/factsheet/en (accessed: 2019-06-01)

[14] https://www.youtube.com/watch?v=iFCKUT5ryQ8 (accessed: 2019-06-01)

[15] A training workshop on system modeling and privacy analysis using CARiSMA in January 2017.

industry partners. In fact our industry partners were able to model their systems and provided three valid system models. The validity of the system models is verified by us in regard to our UML modeling knowledge, and the correctness of the applied UML profiles. It must be noted that we supported them continually in the process of modeling their systems.

The survey execution and our observation regarding the whole process of system modeling include that despite basic knowledge of system modeling, our industry partners modeled their system and subsequently annotated them. The system models provided appropriate case studies, with which we evaluated our proposed privacy analysis methodology in this thesis.

The industry partners reported the results of the evaluation of the VPP (performed by them) in *Deliverable 5.2* [43] of the VisiOn project. We refer to this deliverable and report the archived results (by our industry partners) regarding the modeling tools (particularly CARiSMA). We only report the facts that directly concern the modeling tools (including CARiSMA). Since the DAEM's report on modeling tools is rather abstract, we do not refer to it.

- *Bambino Gesú Hospital (OPBG)*

  > "The CARiSMA modeling tool was used to analyze the design of the system and perform various privacy and security checks."
  >
  > "Despite a lack of basic knowledge of modeling languages such as UML, the administrators have quickly adapted to the tools and believe they can offer a good perspective for analyzing complex aspects of privacy and security [...]."

- *Ministry of Economic Development (MISE)*

  > "[...] the administrators had some difficulties in applying the concepts to a formal model and to represent them in the ways expected by the tools. Also, things were complicated by the fact that many of our users do not have a thorough knowledge of the English language and the tools are only available in English."
  >
  > "[...] many of the modeling strategies that have been applied to the our system can be reused and adapted for other processes, thus improving the documentation and knowledge about other real systems adopted at MISE".

Concerning the results of the case studies, the survey, our observations, and the final reports of our industry partners:

- System modeling, annotating systems with privacy and security issues and eventually performing privacy analysis on the annotated systems were not initially easy tasks for our industry partners. Our observations, however, confirmed that providing training (webinar, workshops) and appropriate materials (manual, screencasts) were key factors to obtain proper system models from our industry partners. Close cooperation of the technical partners (tool owners participated in the VisiOn project) with industry partners with no or little modeling background was required to obtain proper input and conduct a privacy analysis.
- The reports provided by the industry partner express the fact that system design modeling and subsequently performing model based privacy analysis offer proper means to concentrate on aspects such as privacy and security in system development and identify privacy and security violations in IT systems.

Below, we discuss the limitations of our proposed model-based privacy analysis methodology and provide the potential directions for future work.

4.5 Discussion and Limitations

The case studies, to which our privacy analysis is applied, were originally modeled by our industry partners participated in the VisiOn project. Due to their lack of knowledge to correctly model a system using UML and later to appropriately annotate a system model with privacy and security profiles, we supported our partners with providing webinars and workshops. Thus, the resulting system models may reflect our knowledge on system modeling as well. In the future, we aim to study larger practical system models.

Following Section 4.4.2.1, one of the major difficulties of our industry partners was to correctly annotate system models with the privacy and security profiles. Above, we emphasized the support that we offered to assist our partners in annotating their system models. Additionally, in Section 7.2.2.2, we introduce a mechanism to automatically generate help reports out of certain privacy and security requirements, thereby facilitating the process of annotating system models. Our support can be, however, improved by introducing a tool that strives automatic annotation of system models.

Our model-based privacy analysis aims to identify privacy design violations from the early phases, thereby addressing one of the main challenges to operationalize *privacy by design* (see Section 1). Following the running example introduced in Section 2.2, we mentioned that DAEM (the IT company of Municipality of Athens) is in the process of developing a system to service the citizen of Athens. Concerning Section 4.4, our model-based privacy analysis enabled DAEM to identify the privacy design violations given an annotated system model. Additionally, our privacy analysis methodology may be applied to existing systems, which are not in the early phases of development, to identify privacy design violations. In this case, our model-based privacy analysis does not necessarily support operationalizing *PbD*, but it shields one (for instance privacy expert) from the complexity of the system that is analyzed and allows to concentrate on specific aspects such as privacy and security. This require, the annotated system models of existing systems that have to be analyzed.

Analyzing a system model does not guarantee that all violations regarding the privacy preferences will be identified and the real implementation (code) of a system guarantees all privacy preferences. However, identifying the privacy design violations in the early phases of development through analyzing system models, which are available from the onset of development, assists the system developers to integrate recommended privacy and security controls into the system that is analyzed and to accomplish the core of *PbD*.

Following Section 4.3.4, we introduced the details of the *retention check*. We specifically explained that currently in our analysis, we only investigate the existence of appropriate mechanisms that eventually delete or restrict personal data, after it has been processed for a specific purpose. In an activity diagram, using a *timeEvent* action (an hour glass symbol), or generally an *acceptEvent* action (a concave pentagon symbol), it may be possible to verify whether appropriate actions are triggered once in a while or after a specific amount of time. In fact, we may benefit from such symbols to enhance the privacy analysis process, however, currently, such functions are not supported by our proposed methodology.

By performing a privacy analysis on a system model given a set of privacy preferences, a set of analysis results including, specific privacy design violations may arise. Such violations may be derived from wrong or strict specification of the privacy preferences. In other words, the identified privacy design violations are not necessarily the result of ignoring privacy requirements in a system design. In fact, an appropriate mechanism may resolve such violations by negotiating (trade-off). Such negotiations (trade-offs) may benefit from (or be realized by) the PLAs. In the future, we aim to establish a proper mechanism to manage such negotiations.

Since CARiSMA is based on the analysis of the system models that are modeled using UML diagrams, to perform the privacy analysis using the four checks, the systems are modeled using UML. Based on these considerations, in this chapter, the privacy profile is defined for the UML elements. However, concerning the description of each stereotype (Section 4.3.3), the privacy concepts may be adapted for other modeling languages. Furthermore, concerning the fact that IT systems may be modeled using different modeling languages, a transformation may be defined to perform the privacy checks on such models.

Generally, our model-based privacy analysis methodology demands the existence of a set of specific model elements in the system model (such as *dataStore* node) to perform an analysis.

According to Section 2.1, *privacy by design* does not exclude *privacy by default* but gives special importance to the *design* phase. Following Article 25 of the GDPR and concerning [76] and [42], *privacy by default* prescribes that a set of default settings in a system has to be stated ensuring that personal data is only processed for specific purposes defined in compliance with the law. As it is mentioned in Section 3.5.2, by defining specific interval sublattices one may indicate pre-defined (default) ranges in privacy preferences. This supports the principle of privacy by default. For instance, in a purpose lattice, which indicates the set of all possible processing purposes, and before specifying the privacy preferences, an interval sublattice maybe defined (for instance in compliance with a specific regulation), stating that an authorized purpose to process a piece of personal data can be only chosen from this interval sublattice.

4.6 Related Work

Generally, model-based privacy analysis has attracted little attention in the scientific literature so far. A possible explanation is the earlier lack of legal incentives driving its adoption process.

Model-based system design analysis approaches. UMLsec [127, 128] provides an approach to develop and analyze security critical-software, in which security requirements such as integrity, availability, and confidentiality are specified in system models. Moreover, the security analysis techniques have been integrated with the requirements elicitation phase [37, 182]. However, UMLsec analysis does not consider privacy.

Basso et al. [23] provide a UML profile for privacy-aware applications. This profile enables one to describe a privacy policy that is applied by an application and keep track of the elements that are in charge of enforcing the policy. This profile does not enable one to analyze the design of a system.

Kalloniatis et al. [131] propose a method (PriS) for incorporating privacy user requirements into the system design process. PriS provides a methodological framework to analyze the effect of privacy requirements on organizational processes. The authors focus on the integration between high-level organizational needs and IT systems. A privacy analysis is not conducted on a system design.

Colombo et al. [54] introduce MAPaS, a promising model-based framework for the modeling and analysis of privacy-aware systems. MAPaS is built upon PaML—a UML profile to model purpose-based access control systems. They provide a set of analysis functions to assess domain models that are presented as class diagrams. In our methodology, in addition to the authorized purposes, we consider the other three key privacy elements. In [55], the same authors propose an approach (built upon MAPaS) to execute SQL queries based on purpose and role-based privacy policies. Their work is an initial step toward the definition of privacy-aware database management systems (DBMSs). Similar to the MAPaS, they do not cover all four key privacy elements in their work. Alshammari et al. [13] provide a UML profile to express privacy related concepts in the *Abstract Personal Data Lifecycle* (*APDL*) model. APDL model represents the processing of personal data in terms of states, operations, and roles. Their approach provides a promising foundation to support requirement analysis. Likewise, the work presented in [54] (MAPaS), the authors only consider data model diagrams (class diagrams). Such approaches are orthogonal to our approach. In our methodology, in addition to the structure of a system (class diagrams), we particularly focus on the systems' behavior modeled as activities.

Knirsch et al. [137] provide an approach for model-driven privacy assessment in the smart grid. Their approach is built on meta-information, and high-level data flow and assesses the use cases in early design time. Furthermore, Zinke et al. [213] proposed a model to link data privacy requirements with software systems to ensure data privacy compliance. In their work, they introduce a case study to bridge data privacy requirement of retention periods and software systems with the *Information Lifecycle Management* (*IML*). Comparing to our methodology, these approaches are high-level. They do not consider the concrete design of the systems.

Delfmann et al. [63] propose a promising set theory-based pattern matching approach for conceptual models. Their approach is generic and therefore, not restricted regarding its modeling language or application domain. Searching patterns in conceptual models enables one to reveal syntactical errors, and model comparison, and facilitates the improvement of business processes. Moreover, Becker et al. [24] introduce a business process compliance checking approach based on a pattern matching approach. Their approach applies to several conceptual modeling languages and different kinds of compliance rules. They evaluated their approach in a real-world setting. Both approaches focus on business process modeling. These approaches are orthogonal to our approach. Besides verifying a system model regarding a set of specific privacy preferences on the processing of personal data (for instance, specified by a service customer) to identify concrete privacy design violations, a business process model may be checked against a set of compliance rule patterns to identify potential compliance violations.

Abstract and high-level approaches and guidelines to support privacy. In [168, 169], the authors provide a model-based privacy best practice and a variety of guidelines and techniques to assist experts and software engineers to consider privacy when the systems are designed. However, they only focus on top-level security and privacy goals, and do not perform a privacy analysis.

Gürses et al. [96] use two case studies to propose four general steps for applying *privacy by design*. They further argue that the *privacy by design* principle cannot be reduced to a checklist that can be completed without any complexity. Moreover, Spiekermann et al. [189] present concrete guidelines for building privacy-friendly systems. They describe two main approaches for engineering privacy, namely *privacy by policy* and *privacy by architecture*. The former focuses on the enforcement of a set of policies. The latter aims to minimize the collection of personal data and perform anonymization. Their works provide a promising foundation for our methodology, however, their approaches are high-level and cannot be used to analyze concrete system designs with respect to privacy preferences.

4.7 Preliminary Conclusion

We have introduced a modular model-based privacy analysis methodology for industrial ecosystems. The methodology is based on four key privacy elements, namely purpose, visibility, granularity, and retention. A set of stereotypes are introduced to express key privacy elements within the diagrams in a UML system specification. These annotations enable four privacy checks, which adhere to the four key privacy elements. The methodology is integrated into VisiOn project, in which a platform for privacy analysis of public administration systems is provided.

Privacy by design implies that the system's design of IT services has to be analyzed to verify whether the required privacy levels are fulfilled and where necessary appropriate technical and organizational controls must be implemented to support privacy and data protection (Section 4.1). In the following chapters, we describe how we benefit from the results of a model-based privacy analysis to identify privacy risks and proper controls (to mitigate the identified risks).

Chapter 5

Supporting Privacy Impact Assessment by Model-Based Privacy Analysis

This chapter shares material with the SAC'18 paper "Supporting Privacy Impact Assessment by Model-Based Privacy Analysis" [9].

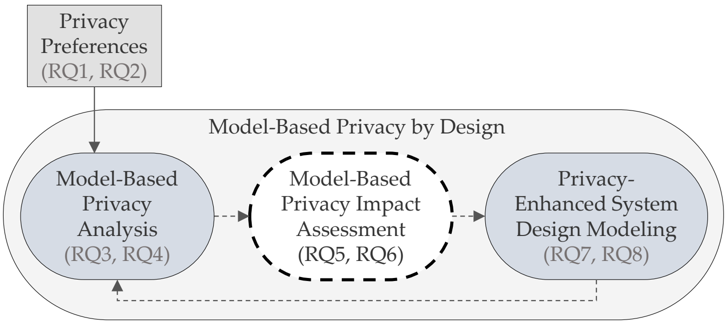

Privacy Preferences (RQ1, RQ2)
Model-Based Privacy by Design
Model-Based Privacy Analysis (RQ3, RQ4)
Model-Based Privacy Impact Assessment (RQ5, RQ6)
Privacy-Enhanced System Design Modeling (RQ7, RQ8)

Figure 5.1: The highlighted section (dashed lines) denotes how Chapter 5 contributes to the overall workflow (introduced in Section 2.3).

In Chapter 1, we mentioned that identifying and mitigating privacy risks is at the core of *privacy by design* (*PbD*). Article 35 of the GDPR [198] requires to conduct a *privacy impact assessment* (*PIA*) to determine the privacy risks. However, existing *privacy impact assessment* (*PIA*) methodologies cannot be easily conducted, since they are mainly abstract or imprecise. Moreover, they lack a methodology to conduct the assessment concerning the design of IT systems. In this chapter, we propose a novel methodology to conduct a *PIA* supported by a model-based privacy and security analysis. In our *PIA* methodology, the design of a system is analyzed and, where necessary, appropriate security and privacy controls are suggested to improve the design. We evaluate our methodology based on three practical case studies of the VisiOn project and a quality-based comparison to state of the art.

5.1 Introduction

Article 35 of the GDPR prescribes *privacy impact assessment* (*PIA*) [198]. A *PIA* aims to conduct a systematic risk assessment in order to identify privacy risks and impose technical and organizational controls to mitigate those risks.

A *PIA* shall be conducted prior to the processing, in the early phases of development. This follows the principle of *PbD* that is stipulated in Article 25 of the GDPR. *PbD* requires that the design of IT systems must be focused or technically adapted, by implementing appropriate controls, thereby ensuring the principles relating to the processing of personal from early phases of system design. In fact, *PbD* encompasses the entire process of *PIA*, namely, identifying the privacy requirements and privacy threats, performing a risk analysis, and choosing proper controls.

Despite the political momentum to establish *PIA*s, and while the governments in Canada, the UK, Australia, and the US conduct *PIA*s in critical sectors, the *PIA* adoption in the IT sector is still rare, particularly in Europe [158]. A possible explanation is the earlier lack of legal incentive. In addition, although a set of legal documents such as the UK *PIA* code of practice [59], and the CNIL's methodology [86] describe the process of conducting *PIA*s; they generally are not suitable to be a process reference model. They describe a set of generic and abstract steps toward *PIA*s, and most importantly, they do not consider the concrete design of a system to identify concrete design violations and threats.

In this chapter, we investigate the following research questions:

RQ5: *Given a system model, how can concrete privacy threats be identified?*

RQ6: *How can a privacy impact assessment be conducted to identify the privacy risks?*

To address these research questions, we make the following contribution:

- We leverage two model-based approaches to analyze system models, namely our proposed model-based privacy analysis (introduced in Section 4.3) and

UMLsec approach (described in Section 4.2.3). Although these two approaches may detect privacy design violations, none of them provides a mechanism to further evaluate the analysis results to identify which harmful activities or threats may be exploited from those privacy design violations. We identify a set of resulting threats, harmful activities, and privacy risks, caused by those privacy design violations (Section 5.3.3).

- To conduct a *PIA*, a set of privacy targets are needed. Privacy targets are derived from the privacy principles. We benefit from the privacy targets introduced by the BSI[1] (German Federal Office of Information Security) *PIA* methodology [159]. However, to fully support the GDPR, we extend the list of privacy targets proposed by BSI with three new privacy targets (Section 5.3.4).
- Our proposed *PIA* assesses the risks (Section 5.3.4) and subsequently suggests proper controls to mitigate the privacy risks (Section 5.3.5).
- We apply our methodology to three industrial case studies from the public administration domain (Section 5.4.1). We provide a comparative evaluation of existing approaches (Section 5.4.2).

The remainder of this chapter is organized as follows. In Section 5.2, the necessary background is provided. In Section 5.3, we explain how *PIA* can be supported by model-based privacy analysis. In Section 5.4, we present our case studies and evaluation. In Section 5.5, we discuss our results. In Section 5.6, we discuss related work. Finally, in Section 5.7, we conclude.

5.2 Background

In this section, we provide the necessary background for this chapter.

5.2.1 Risk, Threat, and Risk Analysis

A glossary including the definition of the key terms used in this thesis is provided in Appendix A. In this section, we refer to ISO 27002 standard [120] and explain the terms that are frequently used when we introduce our proposed *privacy impact assessment* methodology.

[1] `https://www.bsi.bund.de/EN` (accessed: 2019-06-01)

The term *threat* means "a potential cause of an unwanted incident, which may give rise to harm in a system or organization." The privacy design violations (the results of a model-based privacy analysis) yield a set of privacy threats.

The term *risk* means "a combination of the likelihood of an event and its consequence." To estimate the privacy risks, a similar definition is used in this thesis. However, in our proposed impact assessment, we only consider the severities (consequences).

The term *risk assessment* refers to the "overall process of risk analysis." *Risk analysis* means "systematic use of information to identify and estimate risk." A risk assessment includes a systematic approach to estimate the magnitude of risks and comparing the results with an acceptance risk level to determine the significance of the risks. An assessment should be performed periodically to address the possible changes in a system. Generally, the results of an assessment determine a guideline to enhance the assessed system.

5.2.2 Privacy Impact Assessment

Article 35 of the GDPR uses the term *Data Protection Impact Assessment* to prescribe an *"assessment of the impact of the envisaged processing operations on the protection of personal data."* In this thesis, we use the term *Privacy Impact Assessment* instead, for the following rationale: Oetzel et al. [158] argue that privacy is a complex term and is used to appoint different interests, from confidentiality and integrity to transparency and anonymity, therefore, privacy extends beyond the notion of data protection. However, they declare that the privacy threats (identified by Solove [186]) can be addressed through the existing data protection regulation and therefore the two terms can be considered the same (see Section 2.1 for more details).

The *PIA* methodology provided in this work is based on the *PIA* guideline [159] introduced by BSI. In [158], this guideline is extended and introduced as a seven step methodology:

1. System characterization,
2. Specification of privacy targets,
3. Evaluation of the degree of protection demand for privacy targets,
4. Identification of threats,
5. Identification of controls,

6. Implementation of controls,
7. Generation of *PIA* report.

Concerning the steps mentioned above, given a concrete system design, the BSI *PIA* guideline does not specify how we may:

- Identify the concrete threats and calculate the risks arising from those threats,
- Propose proper controls to mitigate those risks and improve a concrete system design.

5.2.3 Privacy Targets

In the BSI *PIA* guideline, to evaluate the degree of protection demand and perform an impact assessment (Step 3, Section 5.2.2), the privacy targets have to be specified in advance (Step 2, Section 5.2.2). Generally, in a risk assessment process, including a *privacy impact assessment*, the aim is to investigate *"what is at risk."* The GDPR introduces a set of privacy principles relating to the processing of personal data (Article 5), which have to be ensured by a system which processes personal data. However, since such principles are semantically more generic than concrete system functionalities, it is difficult to use them for the purpose of assessing a system that processes personal data [158]. A *PIA* has to focus on concrete system characteristics and should identify specific design goals. Therefore, the privacy principles must be translated into concrete and functionally enforceable privacy targets [176, 177].

In our *PIA* methodology, we use the privacy targets introduced by the BSI *PIA* guideline [158, 159]. These privacy targets are mainly derived from the privacy principles formulated in *Directive 95/46/EC*. To fully support the GDPR, we propose to add three new privacy targets to the existing list of privacy targets.

5.2.4 Privacy Threats

A main step in a *PIA* is to identify the privacy threats. Several documents are available to introduce privacy threats. As a source for privacy threats in this thesis, we consider the list of privacy harmful activities by Solove [186]. The document is officially called *a taxonomy of privacy* and lists the privacy threats observed over a century of U.S. legal history.

Solove, in his taxonomy of privacy, aims to focus on different kinds of activities that affect privacy negatively. This taxonomy enables one to shift focus from the broad term *privacy* toward a set of specific activities that are resulting from various violations. Moreover, it is an attempt to understand various privacy harms (often to individuals) that have achieved a notable degree of social recognition and may create problems. In this thesis, following the privacy and security analyses, it is specified how the violations in a system design may cause the harmful activities introduced by Solove.

5.3 Model-Based Privacy Impact Assessment

Figure 5.2 demonstrates the steps of our *PIA* methodology. To support *PbD*, this methodology is to be applied in the early phases of system design. However, it may also be used to conduct a *PIA* on existing systems. For instance, when an existing system is modified, the *PIA* has to be repeated. Iterations of the *PIA* may be conducted on a system along with the system development or due to system modifications.

The first two steps of our proposed *PIA* methodology are the steps that we already discussed in the previous chapter. However, to demonstrate the process of a complete *PIA*, these two steps are shown in Figure 5.2. In the following sections, we describe each step separately.

5.3.1 Systematic Specification of System and its Privacy-Critical Parts

The first step of a *PIA* is to specify the system. In this thesis, a system is modeled by UML. To enable an analysis, a system model is annotated with the privacy (Section 4.3.3) and the security (Section 4.2.3—Background) profiles. The *PIA* methodology proposed in this chapter is not limited to UML. Various modeling languages may be used to model a system. However, this calls for introducing appropriate privacy and security mechanisms to allow expressing privacy as well as security issues in a model and to perform an analysis

In this step, we additionally verify whether conducting a *PIA* is necessary. Following Article 35 of the GDPR [198], a *PIA* shall, in particular, be required in the case of:

- systematic and extensive processing of personal data,

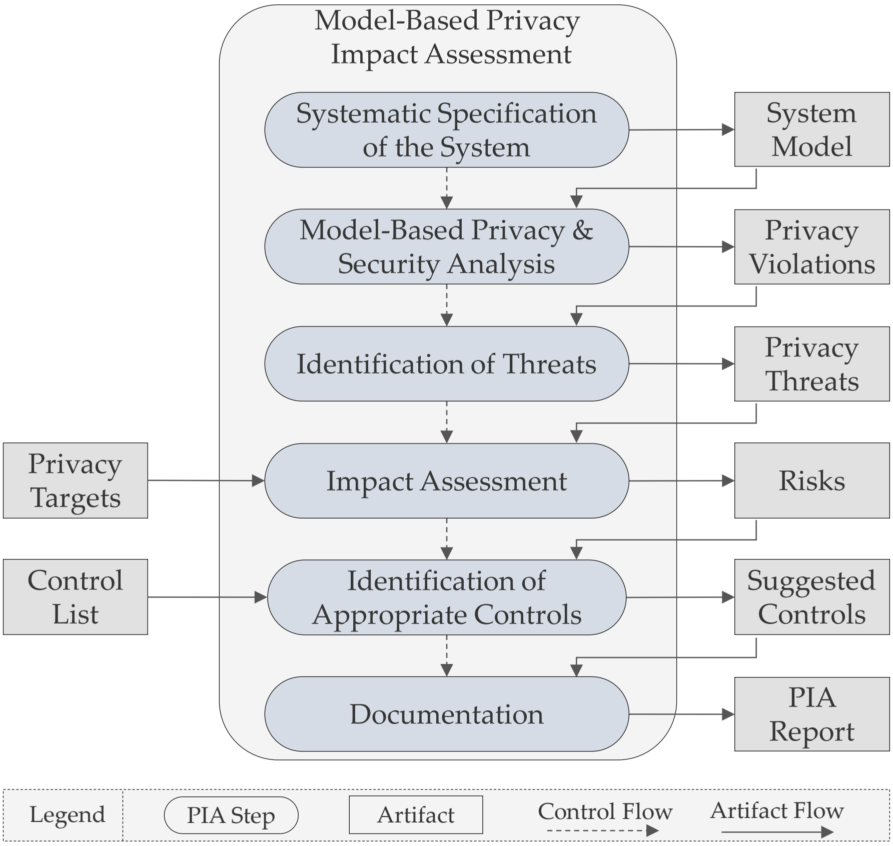


Figure 5.2: *Privacy impact assessment* supported by model-based privacy analysis. Iterations of the *PIA* may be performed along with the system development or due to system modifications.

- Processing of special categories of personal data (defined in Article 9 of the GDPR),
- Systematic monitoring of a publicly accessible area on a large scale.

We benefit from our proposed privacy profile introduced in Section 4.3.3.1 (particularly the «`sensitiveData`» stereotype) to verify whether conducting a *PIA* is necessary. The stereotype «`sensitiveData`» specifies that a *NamedElement* in a UML diagram such as an *ObjectNode* is or contains personal data. An ObjectNode in an *activity diagram* annotated with «`sensitiveData`» specifies that a piece of personal data is processed by performing the associated behavior (represented as the activity diagram) and a *PIA* has to be conducted.

Due to the fact that different categories of personal data are introduced in the GDPR, a categorization of the «`sensitiveData`» stereotype is necessary. The privacy profile supports the categorization of «`sensitiveData`» by defining the tag *category*. The value of this tag may belong to four distinct categories:

- *commonPersonalData* adheres to the definition of personal data prescribed in Article 4 of the GDPR.
- *special* adheres to special categories of personal data stipulated in Article 9 of the GDPR.
- *generalIdNo* adheres to a national identification number or any other identification of general application stated in Article 87 of the GDPR.
- *privacyRelevantData* specifies the data that initially are not considered as personal data, however, later is related to a data subject.

This categorization provides only a baseline for identifying and assessing different categories of personal data; however, concerning different regulations and specific needs of IT systems, other categories can and should be added.

5.3.2 Model-Based Privacy and Security Analysis

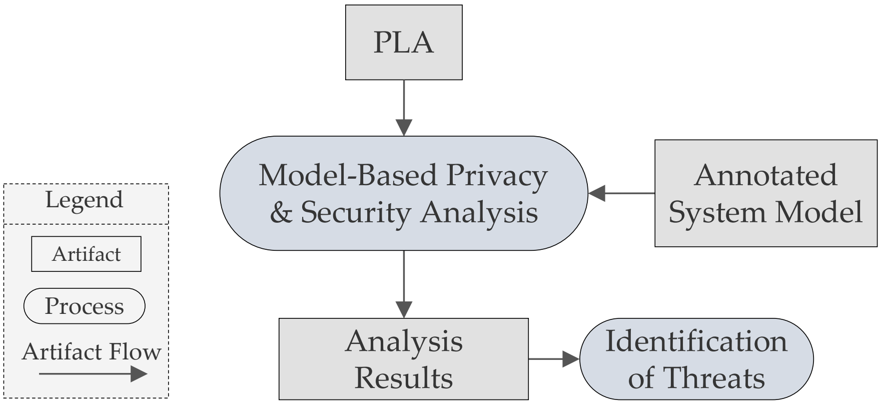


Figure 5.3: The figure shows Step 2 and 3 of the *PIA* methodology. The model-based system analysis includes the model-based privacy analysis methodology proposed in Chapter 4.

In the second step, the system models from the previous step are analyzed regarding privacy preferences and security requirements. The analysis is performed in a model-based manner using a set of privacy and security checks. Figure 5.3 presents the workflow of the tasks performed in the second step. This step includes the model-based privacy analysis methodology that we proposed in Chapter 4. The

Table 5.1: The mapping between harmful activities and the privacy/security checks. The table shows how various privacy and security checks are used to identify harmful activities.

Harmful activities [186]	**Privacy/security checks**
Surveillance	Secure links check (Section 4.2.3.1)
Interrogation	Purpose check, Secure links
Aggregation	Purpose check
Identification	Purpose check, Retention check
Insecurity	UMLsec checks
Secondary use	Purpose check, Retention check
Exclusion	PLA does not exist
Breach of Confidentiality	Granularity check, Secure links check, Crypto FOL-analyzer check [128]
Disclosure	Purpose check, Granularity check, Secure links, Crypto FOL-analyzer check
Exposure	Purpose check
Increased Accessibility	Purpose check, Granularity check, Secure links check, Crypto FOL-analyzer check, Secure dependency check (Section 4.2.3.2)
Blackmail	Secure links check, Crypto FOL-analyzer check, Secure dependency check
Appropriation	Privacy check, UMLsec checks
Distortion	Retention check
Intrusion	Purpose check (precisely verifying *marketing* purpose)
Decisional Interference	Purpose check

security analysis is conducted using the UMLsec checks (Section 4.2.3). Following an analysis, the results specify the design violations in regard to the privacy preferences and security requirements. In the next step, the identified design violations are analyzed to identify the threats.

5.3.3 Identification of Harmful Activities and Threats

The identification of threats is an important step in a *PIA* and basically any known risk assessment methodology. In this step, the analysis results of the previous step are evaluated, and the corresponding threats and harmful activities, that may exploit the privacy design violations, are identified. As mentioned in Section 5.2.4, we consider the harmful activities (threats) introduced by Solove.

Table 5.1 demonstrates a mapping between the harmful activities and the privacy as well as security checks. To each harmful activity (sixteen in total), the corresponding privacy or security checks are mapped. The mapping denotes resulting harmful activities following performing associated checks.

The output of this step is a set of harmful activities resulting from the present design violations.

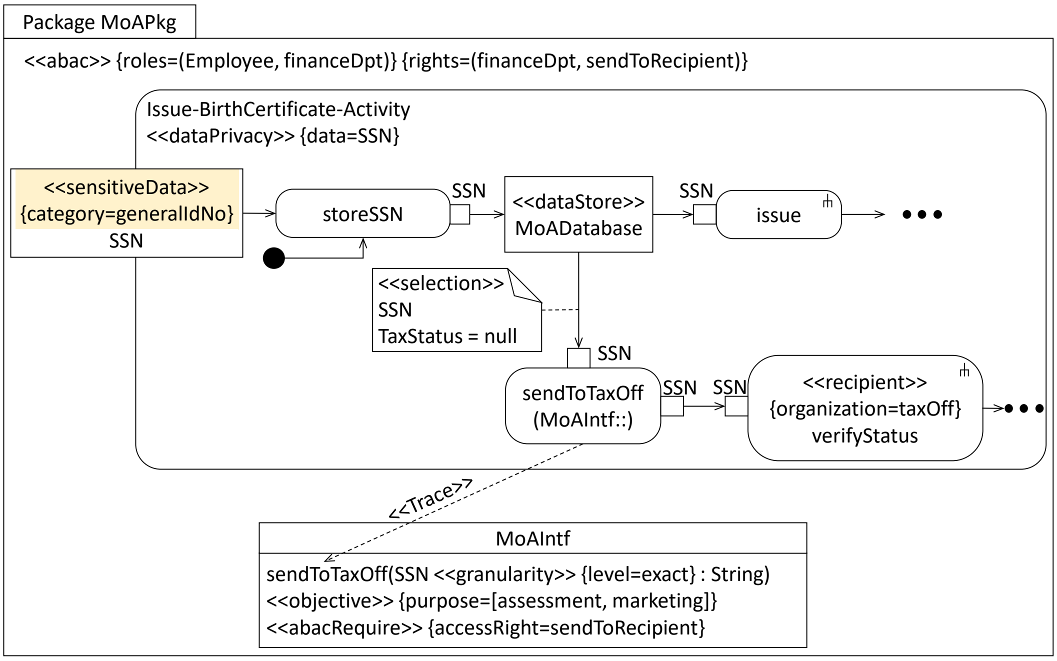

Package MoAPkg
<<abac>> {roles=(Employee, financeDpt)} {rights=(financeDpt, sendToRecipient)}
Issue-BirthCertificate-Activity
<<dataPrivacy>> {data=SSN}
<<sensitiveData>>
{category=generalIdNo}
SSN
storeSSN
SSN
<<dataStore>>
MoADatabase
SSN
issue
<<selection>>
SSN
TaxStatus = null
SSN
sendToTaxOff
(MoAIntf::)
SSN
SSN
<<recipient>>
{organization=taxOff}
verifyStatus
<<Trace>>
MoAIntf
sendToTaxOff(SSN <<granularity>> {level=exact} : String)
<<objective>> {purpose=[assessment, marketing]}
<<abacRequire>> {accessRight=sendToRecipient}

Figure 5.4: Design model excerpt (*issue birth certificate* activity). *SSN* is annotated with «`sensitiveData`» {category = generalIdNo} specifying that the *SSN* is an identification of general application.

For instance, consider Figure 5.4, which demonstrates excerpts from an activity diagram and a class diagram of the *issuing a birth certificate* scenario (the ongoing scenario described in Sections 2.2 and 4.3.4). The activity diagram expresses a business process in which the *SSN* (*Social Security Number*) is processed in MoA (a public administration) and sent to a tax office for verifying the status of the person to whom the *SSN* belongs. *SSN* is annotated with «`sensitiveData`» {category = generalIdNo} specifying that the *SSN* is an identification of general application. Hence, a *PIA* has to be conducted.

We previously defined a set of privacy preferences for the *SSN* using the following notation:

$$PrP_{SSN} = \{(assessment \mapsto (\{financeDept, saleDept\}, partial, 1year))\}$$

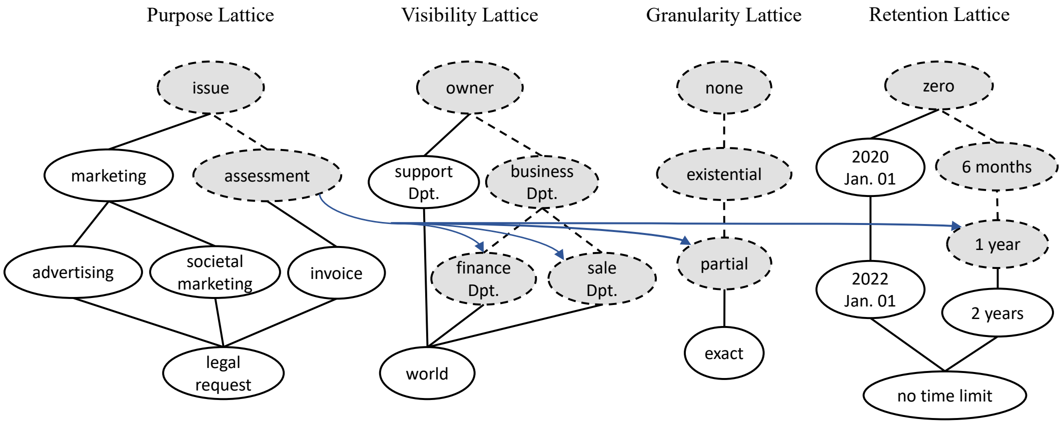


Figure 5.5: The privacy preferences of *SSN* (see Section 3.3). The dashed lines indicate the privacy preferences (authorized purposes, subjects, granularity levels, and retention conditions).

In Figure 5.5, once more, we show the lattices, in which the above-mentioned privacy preferences are specified using dashed lines (Section 3.3). We explained that, performing a privacy analysis, concerning the action *sentToTaxOff*, several design violations are identified. For instance, concerning the operation *sentToTaxOff*—the operation (of a class in the class diagram), to which the *sentToTaxOff* action is traced—«`objective`» specifies that the *SSN* is processed for two purposes, namely *assessment* and *marketing*. However, with respect to the privacy preferences, the *SSN* is not authorized to be processed for the purpose of *marketing*. Hence the following privacy design violation is identified (see Section 4.3.5 for more details on the application of the privacy checks):

> "*SSN* is processed for the unauthorized purpose *marketing*."

Concerning Table 5.1, this design violation in which a piece of personal data is used for unauthorized purposes (precisely marketing purpose), leads to *secondary use* and *intrusion*. According to Solove [186], the former refers to the use of data for reasons unrelated to the initial purposes for which the data is collected. The latter refers to the activities that can disturb one's life, destroy one's solitude and make one feel uncomfortable.

One may use additional scientific sources for the privacy threats, such as the ENISA[2] (European Union Agency for Network and Information Security) threat

[2] `https://www.enisa.europa.eu/` (accessed: 2019-06-01)

landscape [79] or the CSA (Cloud Security Alliance) top threats (*Treacherous 12* [50]), which would then need to be mapped to the checks.

5.3.4 Impact Assessment

In this step, we specify the impact of the identified privacy design violations on a system. To this end, we conduct a risk assessment, in order to identify *"what is at risk."* Generally, a set of privacy targets, which needs to be guaranteed by system design, is required to enable a *PIA*. As mentioned earlier, in our *PIA* methodology, we use the privacy targets of the *PIA* guideline proposed by BSI [158, 159].

Following [158], the proposed privacy targets in the BSI guideline provide only a baseline, and more targets can be added. Concerning the GDPR, we propose three new privacy targets:

P1.9 Ensuring the categorization of personal data: We require to specifically indicate the categorization of a piece of data as a privacy target. Different categories of personal data are stipulated in the GDPR. In Section 4.3.3.1, we introduced four categories of personal data.

P1.10 Ensuring the prevention of discriminatory effects on natural persons (fairness): Following the GDPR (p. 14):

> "[...] discriminatory effects on natural persons on the basis of racial or ethnic origin, political opinion, religion or beliefs, trade union membership, genetic or health status or sexual orientation, or that result in measures having such an effect, have to be prevented."

Therefore, we propose a privacy target to ensure the prevention of discriminatory effects on natural persons.

P6.3 Ensuring the effectiveness of technical and organizational measures: According to Article 32, paragraph 1.(d) of the GDPR:

> "[...] a process for regularly testing, assessing and evaluating the effectiveness of technical and organizational measures for ensuring the security of the processing is needed."

Therefore, we propose a privacy target to ensure the effectiveness of technical and organizational measures.

To assess the impact of privacy design violations, the identified threats, and the harmful activities on the privacy targets, we extend the mapping provided in Table 5.1 by the privacy targets. Table 5.2 shows the mapping between the privacy targets and the privacy and security checks. Oetzel et al. [158] provide a relationship between the Solove's harmful activities and the BSI privacy targets. The Oetzel et al.'s relationship is judged by three independent privacy experts. We benefit from this relationship, and in Table 5.2, we indicate how we may identify the privacy targets at risks with respect to our proposed privacy analysis methodology and the UMLsec checks. The second column of the table presents the analysis means, mainly including privacy and security checks, to identify the privacy targets at risks. The complete mapping between the analysis means, the harmful activities [186], and the privacy targets is provided in Appendix D[3].

In this table, not all the privacy targets are mapped to specific checks. For instance, *P1.1* and *P6.3* are mapped to *Existence of PLA*. In this case, the existence of PLA or a specific section in a PLA, ensures the corresponding privacy targets.

P1.10 ensuring the prevention of discriminatory effects on natural persons (fairness) is currently mapped to *Existence of PLA*. In [174], Ramadan et al. proposed a methodology to support discrimination analysis relying on system models and available data. This methodology is based on UML and may be used to verify *P1.10*.

Some of the privacy targets are mapped to one specific check. For instance, *P1.2, P1.4*, and *P5.2* are mapped to the *purpose check*. In this case, the check is used differently to realize the mapping. For example, in the case of *P1.2*, it has to be verified whether for a piece of personal data, a legitimate purpose(s) in a PLA is specified. For *P1.4*, it has to be verified if a piece of personal data is processed for unauthorized purposes. For *P5.2*, it has to be verified if a piece of personal data is processed particularly for *marketing* purposes.

For some privacy targets, a new utility of a check is defined to realize the mapping. For instance, *P4.3* refers to Article 20 of the GDPR and implies that a data subject (who provides data) shall always have the access right to his/her personal data. The visibility check is used to verify this (see Section 4.3.4). It has to be verified if an access right to a piece of personal data for a data provider during the whole processing exists.

[3] The mapping between the privacy targets and the Solove's harmful activities is based on the existing work by Oetzel et al. [158]. We extended their work (mapping) with our three new proposed privacy targets and a mapping to the analysis means.

Some privacy targets are mapped to a stereotype or a tag, which belongs to the privacy profile. For instance, *P5.2* is mapped to «`objective`» stereotype. In this case, it has to be verified that for an operation, which processes a piece of personal data, an *objective* (*purpose*) is defined.

By evaluating the results of an analysis and using Table 5.2, we identify which privacy targets are at risk. For instance, considering the example provided in Figure 5.4, the *SSN* is used for an unauthorized processing purpose:

> "*SSN* is processed for the unauthorized purpose *marketing*."

Concerning Table 5.2, two targets, namely *P1.4* and *P5.2*, are at risk.

After identifying *"what is at risk,"* a privacy risk assessment has to be performed. In Section 5.2.1, it is mentioned that a risk is a combination of the likelihood of an event and its consequence [120]. A privacy risk assessment can be performed differently, provided that a likelihood and a severity are obtained for each risk [86].

In privacy domain, calculating the likelihood that a threat may occur, is fuzzy. The authors of [158] discourage the use of threat likelihood estimation from security risk analysis in the privacy domain, because privacy is related to human emotions [186] and if a human right such as privacy is threatened, the arising risks have to be mitigated. The CNIL *PIA* methodology [86, 87] introduces a scale to estimate likelihoods of threats, and provides a set of variables that affect this scale, such as *opening on the internet, data exchange with third parties,* and *variability of the system*. In our methodology, estimating such likelihoods is not essentially relevant. The systems that we analyze are mostly open on the internet, include data exchange with third parties and are interconnected with other systems. Thus, the CNIL likelihood variables cannot be used to estimate the likelihoods.

In our *PIA* methodology, similar to the Oetzel et al.'s PIA methodology [158], we do not consider the specific likelihoods and if a privacy threat exists, we control it. Particularly, our risk assessment is only based on the severities of the identified privacy design violations on the privacy targets. If a privacy target is at risk, we predict the potential impact of this risk. This depends on two factors:

- What kind of personal data is analyzed?
- What kind of system is analyzed?

To assess these two factors, and consequently assess the risks, we introduce *Personal Data Category Value (PDCV)* and *Impact Value (IV)*. The former addresses the

Table 5.2: The relation between the privacy targets and the privacy/security analysis

Privacy target	**Analysis means**
P1.1 Ensuring fair and lawful processing by transparency	Existence of PLA
P1.2 Ensuring processing only for legitimate purposes	Purpose check
P1.3 Providing purpose specification	«`objective`»
P1.4 Ensuring limited processing for specified purposes	Purpose check
P1.5 Ensuring data avoidance	«`dataPrivacy`»
P1.6 Ensuring data minimization	Purpose check
P1.7 Ensuring data quality, accuracy and integrity	UMLsec checks
P1.8 Ensuring limited storage	Retention check
P1.9 Ensuring the categorization of personal data	*category* tag
P1.10 Ensuring the prevention of discriminatory effects on natural persons (fairness)	Existence of PLA
P2.1 Ensuring legitimacy of personal data processing	Privacy check
P2.2 Ensuring legitimacy of sensitive personal data processing	Privacy check
P3.1 Adequate information in case of direct collection of data	Existence of PLA
P3.2 Adequate information where data is not obtained directly	Existence of PLA
P4.1 Facilitating the provision of information about processed data and purpose	Privacy check
P4.2 Facilitating the rectification, erasure or blocking of data	Retention check
P4.3 Facilitating the portability of data	Visibility check
P4.4 Facilitating the notification to third parties about rectification, erasure and blocking of data	Retention check
P5.1 Facilitating the objection to the processing of data	Privacy check
P5.2 Facilitating the objection to direct marketing activities	Purpose check (marketing purpose)
P5.3 Facilitating the objection to data-disclosure to others	Visibility check
P5.4 Facilitating the objection to decisions on automated processing	Existence of PLA
P5.5 Facilitating the data subjects right to dispute the correctness of machine conclusions	Existence of PLA
P6.1 Ensuring the confidentiality, integrity, availability, and resilience	Security checks
P6.2 Ensuring the detection of personal data breaches and their communication to data subjects	Purpose check (notification purpose)
P6.3 Ensuring the effectiveness of technical and organizational measures	Existence of PLA
P7.1 Ensuring the accountability	Existence of PLA

criticality of a piece of personal data. The latter specifies the protection demand concerning each privacy target. Afterwards, we provide a formula to estimate the potential impact of a risk.

Personal Data Category Value (PDCV):

A categorization for «`sensitiveData`» is presented in Section 5.3.1. This categorization affects our severity estimations. For instance, consider the two cases:

I A privacy target is at risk, since a piece of personal data from the *special* category is used for unauthorized purposes.

II A privacy target is at risk, since a piece of personal data from the *commonPersonalData* category is used for unauthorized purposes.

The severity in case **(I)** is higher than in case **(II)**. To estimate the effect of different personal data categories on the estimation of severities, we introduce *Personal Data Category Value (PDCV)*. A *PDCV* is used to evaluate the criticality of a piece of data in a specific personal data category. In Table 5.3, three different values are assigned to the four categories. The criticality of the two categories *special* and *generalIdNo* are equal and evaluated to the highest value.

Following Section 5.3.1, the categorization of sensitive data (personal data) in this thesis is not exhaustive, and it may be extended concerning various factors. Moreover, the reason not to differentiate between the criticality of the two categories, *special* and *generalIdNo*, follows from the fact that the GDPR precisely uses two different articles to specify these two categories. However, from the GDPR, we could not ascertain the criticality of these two categories. Therefore, adhering to the GDPR, two separate categories are defined to cover these two articles, but we assume that the criticality of a general identification number (*generalIdNo*) equals to the criticality of special categories of personal data (*special*). Based on different contextual factors, this assumption may vary as well.

Table 5.3: Personal Data Category Values (*PDCV*s)

PDCV	Personal data category
0.25	privacyRelevantData
0.5	commonPersonalData
1	special, generalIdNo

The ranking of the criticality of a piece of personal data (*PDCV*) showed in Table 5.3, follows the ranking proposed by ENISA for the *Data Processing Context (DPC)* in [60]. ENISA uses *DPC* to evaluate the criticality of a given data set in a specific processing context by a value between 1 and 4. Similar to the work by ENISA, the values assigned to different personal data categories (*PDCV*) are in fact basic values to be seen just as an evaluation of the criticality related to the corresponding categories. In other words, these values are introduced for the purpose of impact assessment in this thesis and are not to be seen as a general ranking for the personal data categories.

Impact Value (IV):

Different perspectives of different stakeholders in different systems may affect the severities. Thus, we need to evaluate the degree of protection demand for each privacy target that is at risk, from a perspective of a stakeholder. We focus particularly on two stakeholders, namely a data subject that provides the data (data controller) and the data processor that performs the processing of personal data. Following [158], the reason to consider these two stakeholders is that in case of an unauthorized personal data processing, both a data subject that provides data (data controller) and the data processor that performs the processing are damaged.

To estimate the degree of protection demand for each privacy target, following an analysis, we generate several questions based on the privacy targets and ask for the feedback of data subjects and data processors. Table 5.4 introduces the *Impact Value (IV)* to evaluate such feedback. In fact, two impact values are calculated regarding the answers that each stakeholder provides, namely *Data Controller IV (DC-IV)*, *Data Processor IV (DP-IV)*. *DC-IV* is associated with the *public embarrassment* and *DP-IV* is associated with the data processor's *reputation*.

Final Impact Assessment Score (IA):

A final score of an *Impact Assessment (IA)* for each privacy target that is at risk is calculated by multiplying all the three values *PDCV*, *DC-IV*, and *DP-IV*:

$$IA = PDCV \times DC\text{-}IV \times DP\text{-}IV$$

Each of these three values reduces or increase the final impact score. Therefore, the combination of the three values *PDCV*, *DC-IV* and *DP-IV* (multiplication) gives the final *IA* score for a privacy target at risk.

The formula to calculate the overall *IA* score follows the risk estimation methodology introduced by ISO 27005 standard [121], the methodology to assess the severity of personal data breaches introduced by ENISA [60] and the Oetzel et al.'s privacy impact assessment [158]. In ISO 27005 standard, the assessment of the severities is

Table 5.4: Impact Values

Impact Value (IV)	**Impacts**
1-Negligible	Either not affected or may encounter a few inconveniences
2-Limited	May encounter significant inconveniences (may be able to overcome)
3-Significant	May encounter significant consequences (may be able to overcome with difficulties)
4-Maximum	May encounter irreversible consequences (may not overcome)

based on business impacts taking into account the loss of confidentiality, integrity and availability of the assets. This, in fact, corresponds to the concept of *impact value*s which are estimated concerning the loss of reputation and public embarrassment. Moreover, as mentioned earlier, ENISA to assess the severity of personal data breaches, considers *Data Processing Context*, which corresponds to the criticality of personal data captured by *PDCV*. Furthermore, Oetzel et al. consider several values for the protection demands of two stakeholders in five different scenarios (reputation and financial situation of a data processor, and reputation, financial situation and personal freedom of data controller) to estimate the severities. They eventually consider the highest value of protection demand in the five scenarios to estimate a final value for the severity of a threat.

The analysis of the system model provided in Figure 5.4 identified that two privacy targets *P1.4* and *P5.2* are at risk. Two questions must be generated and asked for the feedback of the stakeholders. For instance, concerning *P5.2*, the question "What would happen when facilitating the objection to direct marketing activities is at risk?" will be generated. Together with this question the concrete detailed information "*SSN* is used for the purpose of marketing" will also be generated. Assume that the two stakeholders evaluate the impacts as following: *DC-IV* is *maximum*, and *DP-IV* is *maximum*. Considering the fact that *SSN* belongs to the category *generalIdNo*, the impact assessment score for this privacy target equals to sixteen ($1 \times 4 \times 4 = 16$).

Eventually, after calculating the *IA* scores for all the privacy targets that are at risk, the identified risks have to be mitigated. The mitigation is performed by the controls that are introduced in the next step. To allow this mitigation and choosing the proper controls, first a categorization for the *IA* scores have to be provided. In Table 5.5, a categorization of different ranges of *IA* score into four categories, namely *low*, *medium*, *high* and *very high* is presented.

To establish the basis of the impact values (*IV*s) and the scales that are shown

Table 5.5: The categorization of the IA Scores

IA Score range	Category
$IA < 4$	Low
$4 \leq IA < 8$	Medium
$8 \leq IA < 12$	High
$12 \leq IA$	Very High

in Table 5.4—to estimate the protection demands with respect to the privacy targets—and to categorize the *IA* scores (Table 5.5), we leverage the definitions, scales, and the categorizations that are provided in the CNIL *PIA* methodology [86, 87].

5.3.5 Identification of Appropriate Controls

An important step in a *PIA* methodology is to identify and recommend appropriate privacy and security controls to mitigate the risks and improve the system design. In our *PIA* methodology, we provide a catalog of privacy and security controls. This catalog is based on the security controls of ISO 27001 [119], the privacy control catalog of NIST [151], the measure catalog of the German IT baseline protection [39], and the privacy strategies that are provided in [108]. The details of this catalog are described in Section 7.3.3.

In order to identify an appropriate set of controls, we mapped the controls in the control catalog to the privacy targets and the security requirements. An excerpt of this mapping with the focus on privacy targets and the NIST controls is shown in Appendix E (Table E.1). Following the identification of the privacy targets that are at risk, using this mapping, we identify a set of controls that potentially mitigate the risks.

The controls are divided into technical and organizational controls. The technical controls explicitly specify which mechanisms have to be incorporated into the system design in order to mitigate the identified risks. An encryption algorithm or an access control are two examples for technical controls. Organizational controls are mainly management or administrative recommendations. For instance, if a privacy target is at risk since no authorized purposes are specified for a piece of personal data in a relevant section of a PLA, an organizational control recommends to conclude a proper PLA with the purpose specification for personal data.

According to Oetzel et al. [158], the strength of the controls in mitigating the risks may vary. Therefore, following [158], we introduce different *levels of rigour* for controls, to express their strength. Oetzel et al. define three levels of rigour. Since in

Table 5.5 four ranges for the impact assessment (*IA*) score are defined and the levels of rigour have to cover the four ranges of *IA*, we define four *levels of rigour*, namely *sufficient*, *medium*, *strong*, and *very strong*. If the *IA* score for a privacy target is *very high*, a control from the category *very strong* is suitable to mitigate the identified risk.

To appropriately mitigate a risk, different factors have to be considered. In Chapter 6, we describe how the suggested controls can be integrated into a system model to adequately mitigate the identified privacy risks.

5.3.6 Privacy Impact Assessment Report

Identifying a common mechanism to report a *PIA* is rather challenging. In [209], the authors state that it is difficult to find published examples of *PIA* reports and the companies that perform a PIA, normally, do not reveal their process. In their work, they analyze a number of existing published PIAs and propose criteria to assess the effectiveness of a *PIA* report. We propose to use privacy level agreements (PLAs) to report PIAs. As mentioned earlier, a PLA aims to specify privacy levels that must be respected by a data processor.

In a PLA, several sections specify different aspects, such as generic information on a data processor, privacy preferences, and security requirements [49]. We compare the current structure of a PLA proposed by Cloud Security Alliance with the criteria introduced in [209] and indicate how a PLA must be extended:

1. **A *PIA* report has to specify if a *PIA* is performed in the early phases of a system development**. Since, in our proposed methodology, the assessment starts in the design phase, it is ensured that we start in the early phases.
2. **A *PIA* report has to specify who conducted a PIA**. Such information are included in the generic section of a PLA.
3. **In a *PIA* report, any relevant information on the processing of a piece of personal data must be specified**. In a PLA, leveraging «`sensitiveData`» and regarding the corresponding activity diagrams, we specify the personal data that is processed, together with relevant information including purpose, visibility, retention, and granularity.
4. **In a *PIA* report the process of an assessment has to be described**. In a PLA, following the six steps of our methodology and the output of each step, we specify several sections to document the artifacts including threats, harmful activities, risk assessments, and proposed controls. The current structure of

a PLA already includes a section on security and privacy controls, however, concerning a *PIA*, the identified controls have to be mapped to concrete system privacy design violations and threats.

5. **A *PIA* report must be published**. Since a PLA that is concluded between two parties must always be available, this criterion is fulfilled.

5.4 Case Studies and Evaluation

In this section, we describe how our proposed *privacy impact assessment* methodology introduced in this chapter may be applied to the case studies of the VisiOn project introduced in Section 4.4. Furthermore, we provide a comparative evaluation to three existing *PIA* methodologies.

5.4.1 Case Studies

To evaluate our *PIA* methodology, the three case studies of the VisiOn project are used. Three public administrations (PAs) modeled their systems using UML. They further annotated their system models with the privacy and security profiles. In the previous chapter, Table 4.4 presented information on the size of the system models and the number of annotations. The example discussed in Section 5.3.3 showed an excerpt of a PA's system model.

The CARiSMA tool (see Section 7.2) is used to perform privacy and security analyses on the three system models. CARiSMA enables different privacy and security checks [6]. After analyzing the system models in each case study, using Tables 5.1 and 5.2, the privacy design violations, respective harmful activities, and privacy risks are identified.

To calculate the corresponding *impact assessment* (*IA*) scores, the feedback of the stakeholders, namely, the experts of each PA system and the customers, concerning the privacy targets that are at risk, are required. The VisiOn privacy platform includes different tools. One of the tools provides a set of questionnaires to obtain the privacy and security needs of service customers and PAs. The questions on the privacy targets may be integrated into these questionnaires. This allows us to initially estimate the protection demands for the privacy targets from two perspectives: PA system experts and the customers and specify the *data processor impact values* (*DP-IV*s) and *data controller impact value*s (*DC-IV*s) for all privacy targets. After performing a model-based analysis, concerning the analysis results, the *impact* *value*s (*IV*s) and the criticality of personal data processed in the each system model, the *impact assessment* (*IA*) scores for the privacy targets at risks are calculated.

Concerning Figure 5.3 and our discussion in Sections 5.3.3 and 5.3.4, following performing a *PIA*, besides the two privacy targets *P1.4* and *P5.2*, two more targets were at risk: namely, *P1.8 Ensuring limited storage* and *P4.2 Facilitating the rectification, erasure or blocking of data*. These two privacy risks emerged since the *SSN* is stored in a database, but no appropriate mechanism was available to remove or restrict them.

Following calculating the *IA* scores, concerning the categorization of *IA* scores in Table 5.5 and the list of NIST controls provided in Table E.1, several controls are suggested mitigating the emerging risks.

Concerning the explored research questions:

RQ5: Given a system model, how can concrete privacy threats be identified? Using model-based privacy and security analyses and the mapping between harmful activities and the privacy/security checks (introduced in Table 5.1), we described how concrete harmful activities (threats) may be identified given a system design. Two model-based approaches, which are supported by a tool (CARiSMA), perform the security and privacy analyses to identify the respective harmful activities and threats.

RQ6: How can a privacy impact assessment be conducted to identify the privacy risks? We introduced a six-step methodology to conduct a *PIA*. Our proposed *PIA* methodology is supported by analyzing system models and identifying concrete privacy design violations. An extended list of privacy targets and an assessment method based on the criticality of processed personal data as well as the feedback of two certain stakeholders are used to determine the privacy risks. In Section 7.3, we introduce the ClouDAT[4] framework [5, 12, 195], which supports one to conduct our *PIA* methodology.

In Section 5.5, we discuss the limitations of our proposed *PIA* methodology and provide potential directions for future work.

5.4.2 Comparative Evaluation

In order to validate the effectiveness of the proposed *PIA* methodology and to verify how this methodology may support existing *PIA* methodologies, we compare this methodology to three recognized *PIA* methodologies in Europe, namely the

[4] `http://www.cloudat.de/` (accessed: 2019-06-01)

Table 5.6: The enhanced support of our proposed *PIA* methodology compared with other three *PIA* methodologies (the UK *PIA* code of practice [59], the CNIL *PIA* methodology [86], and the BSI *PIA* methodology [158, 159]).

Quality criteria for a best practice *PIA* process		Our proposed *PIA*
1	Early start	○
2	General description of the project	✗
	Information flows	✓
	(Other) privacy implications	○
3	Stakeholder's consultation	○
4	Risk assessment	✓
	Risk mitigation	✓
5	Legal compliance check	○
6	Recommendations and action plan decision	✓
	Implementation of recommendations	✓
	PIA report	○
7	Audit and review	○

✓: enhanced support compared to the 3 approaches
✗: not supported
○ : similar to other three approaches

UK *PIA* code of practice [59], the CNIL *PIA* methodology [86], and the BSI *PIA* methodology [158, 159], with respect to the seven *PIA* quality criteria published in [207]. Concerning each criterion, Table 5.6 demonstrates whether our proposed *PIA* methodology supports the other three methodologies (marked by ✓), whether it is similar to them (marked by ○) and whether it fails to support them (marked by ✗).

1. Early start: All four methodologies are used to conduct a *PIA* from the early phases of a system development.

2. Project description: We do not require to explicitly describe the context of a project such as organizational goals. The three *PIA* methodologies (UK, CNIL and BSI) require a general description of the project. Regarding *information flows*, in our proposed PIA, using system models (activity diagrams) we specify how a piece of personal data is processed. The UK *PIA* code of practice records the information flows in whichever format (textual descriptions, or models such as flowcharts), however, such models are not technically analyzed. The CNIL and the BSI methodologies describe the processes (information flows) only generically.

3. Stakeholder consultation: All four *PIA* methodologies support stakeholder consultation. Although we do not require a general description, we support the perspectives of the stakeholders during impact assessment.

4. Risk management: The UK *PIA* code of practice only provides a set of generic risks. In CNIL and BSI, several templates and guidelines are provided to perform a risk assessment and mitigate the identified risks. However, these templates and guidelines are rather imprecise and abstract. We conduct a risk assessment regarding the identified privacy design violations, threats, and the categorization of personal data.

5. Legal compliance check: All four *PIA* methodologies comply with legal requirements and principles. In our work, we updated the list of privacy targets, regarding the privacy principles in the GDPR and added three new targets.

6. Recommendation and report: All four methodologies provide recommendations to adapt and improve their systems. Chapter 7 of the UK *PIA* code of practice, the BSI IT baseline protection [39] and the CNIL *PIA* knowledge bases [87], provide a set of privacy controls to mitigate the identified risks. Similar to these controls, we also provide a list of privacy controls. We choose the proper controls according to the conducted risk assessment. Furthermore, we use PLAs to document a *PIA* report and generate a structured document to include the results of each step of our *PIA* methodology.

7. Audit and review: This criterion requires that a *PIA* report has to be externally audited. By documenting a *PIA* report in a PLA, a formal description of a *PIA* report is generated, which facilitate the external audit of a *PIA* report.

The results of the comparison include that although the UK *PIA* code of practice, the CNIL *PIA* and the BSI *PIA* methodologies support the seven quality criteria, they are rather abstract and generic. They do not perform a system level privacy analysis. Moreover, our *PIA* methodology is supported by a tool (CARiSMA) to perform an analysis.

5.5 Discussion and Limitations

The *privacy impact assessment* methodology (particularly the calculation of the impact scores) is enabled by a set of categorizations of the personal data and impact values and the corresponding values of various categories. As mentioned previously, these categorizations and rankings with different values only provide a baseline for our proposed methodology and can be extended or adapted concerning various factors. Generally, we follow the guidelines and legitimate methodologies to conduct a *privacy impact assessment*.

Our *PIA* is only based on the severities of the identified design violations and their impacts on the privacy targets. The likelihoods of occurring privacy design violations are principally ignored. In [210], the authors propose a generic risk assessment model to generate risk likelihoods. Their approach determines the risk level of assets (an asset can be a piece of personal data) and each risk propagation path. Moreover, the approach assists the decision makers by recommending controls to mitigate the risks. Such an approach can be used to revise our impact assessment (explained in Section 5.3.4) by considering likelihoods to estimate risks. However, following [158], we principally ignore likelihoods, due to the fact that privacy is related to human rights and if it is threatened, the identified risks have to be mitigated.

The evaluation of the proposed *PIA* methodology is partly based on the case studies that are modeled during VisiOn project. Since the public administrations are supported by us to model their systems, the resulting risks are not exhaustive. In the future, we plan to perform more comprehensive impact assessment based on more complicated system models[5].

As mentioned in Section 4.5, through analyzing system models, only design violations are identified. Therefore, our proposed *PIA* methodology is not able to identify all privacy risks of a system. However, the identified privacy risks—following performing our proposed *PIA* methodology—specify which privacy controls have to be integrated into a system in the early phases of development. This, in fact, adheres to the *privacy by design* principle and the first *PIA* quality criterion introduced by Wright et al. [207]

In Table 5.2, the privacy targets are mapped to a set of analysis means, with which we determine the privacy targets at risk. The mapping concerns mainly the proposed privacy checks in this thesis and the UMLsec checks. This mapping only provides a baseline to enable a *PIA* performed based on a system model in the early phases of system design. The analysis means column in this table may be enhanced by further proper model-based methodologies. For instance, as mentioned earlier, Ramadan et al. [174] propose a methodology toward a model-based discrimination analysis. This discrimination analysis benefits from the UMLsec and our model-based privacy analysis methodology. It could be used to identify the privacy design violations related to the new proposed privacy target on fairness; *P1.10 ensuring the prevention of discriminatory effects on natural persons (fairness)*.

In our *PIA*, to calculate *impact assessment* (*IA*) scores, two factors are considered:

[5] In the course of our current EU project (Qu4lity project, EU's Horizon 2020 Program, No. 825030).

the criticality of personal data that is processed and the protection demands from the perspective of two stakeholders. More factors may be considered to estimate *IA* scores. For instance, as mentioned earlier (in Section 5.3.4), Oetzel et al. take into account five different scenarios to estimate protection demands. ENISA consider specific circumstances of a data breach (for instance, by quantifying the loss of confidentiality or any involved malicious intent) to estimate the severity of data breaches. Since such circumstances are only present in particular situations, ENISA adds the values of the quantified circumstances to the final values of the severity of data breaches. Moreover, we may take into account different weights for the critical factors (*PDCV* and *impact value*s). Our formula to calculate the *IA* scores may be modified or adapted in different domains and for different purposes.

Moreover, in the formula to calculate *IA* scores, a multiplication of the values provides the final score. To evaluate this formula, we considered several examples, for instance, by doubling one value and halving another value, where we got the same final score (and consequently same *IA* range). Various risk assessment and severity estimation approaches use the multiplication, the sum, or a combination of these two operators to estimate risks and severities. For instance, ISO 27005 standard states that a final risk level is calculated by multiplying the operands (severities and likelihoods). Butler's framework takes into account the sum of different normalized values in its risk assessment. Oetzel et al. in their *PIA* only consider the highest protection demand to estimate the severity of a threat. ENISA's formula to assess the severity of data breaches is based on both multiplication and sum of various factors. When calculating risks or severities based on the sum of several values, the values have to be normalized.

5.6 Related Work

The approach described in this chapter is a novel methodology to support a *PIA* by model-based approaches. Our work is motivated specifically by Article 35 of the GDPR.

In [45], the authors provide a definition of a PIA and identify the main characteristics that distinguish a *PIA* process from other procedures. Furthermore, a list of criteria to evaluate the *PIA* guidance documents is presented, which is applied to several guidance documents published by government agencies. The work presented in this paper does not provide a concrete methodology or guideline to conduct a PIA.

Methodologies to support or conduct *privacy impact assessment*. In [158], a systematic methodology for *privacy impact assessment* by formally representing a structure to analyze privacy requirements and assisting practitioners to handle the complexity of privacy regulations, is provided. In [34], a process for data protection impact assessment under European general data protection regulation is provided. In [208], the authors review the existing *PIA* methodologies, conduct a survey on *PIA* in the EU, and recommend an optimized *PIA* framework to the European Commission. These approaches are orthogonal to our approach: they describe a *PIA* process, including different steps and provide guidelines to conduct each step. However, in these works, the authors do not provide a methodology to analyze a concrete system design to identify concrete privacy design violations. We propose to use system models to identify concrete design violations and conduct assessments concerning those violations.

In [61], Joyee De et al. introduce a methodology PRIAM (Privacy RIsk Analysis Methodology) to assess the privacy harmful activities and conduct risk assessment. In their methodology, they first gather all required information for a *PIA* (phase I). Afterwards they conduct a systematic privacy risk analysis (phase II). Their work analyzes a system design expressed by a high-level data flow diagram. In our methodology, we analyze the concrete structure and behavior of a system. The data flows in our methodology are expressed by activity diagrams, which manifest the actions and operations (of a system) that process a piece of personal data.

Butler [40] presents a promising framework for conducting risk assessments based on multi-attribute analysis. Multi-attribute analysis [212] is used to evaluate decision alternatives when the decision outcomes are uncertain. The consequences of threats are called attributes. In a multi-attribute risk assessment to estimate the risks a vector of attributes is established, where the value of an attribute indicates the level of damage. In Butler's framework, the consequences of threats are called attributes, and the multi-attribute methods in risk assessment are used to establish a vector of attributes, where the value of the attribute is the level of damage and to estimate the risks. For instance, a threat may cause lost revenue, public embarrassment and lost reputation. The values of these attributes are used to estimate the risks. To estimate the severities of privacy design violations, in this thesis, we considered *impact value*s. In fact, the *impact value*s are similar to the concept of attributes used in Butler's framework to indicate the level of damage.

Meis et al. [143, 144], provide a method to systematically elicit the needed information for a *PIA* from a given set of functional requirements. They use class diagrams to create problem frame models capturing system requirements and their relation to the system environments. They leverage a UML privacy profile [26, 56] to model privacy requirements. Their approach focuses on the system requirements and serves as a starting point for a *PIA*. In contrast, we conduct a *PIA* by analyzing a system and assessing privacy risks concerning a set of privacy targets.

Identifying and analyzing privacy threats. Deng et al. [65] provide LINDDUN, a methodology to model privacy specific threats, by introducing a list of privacy threat types and a mapping to the elements of a system. Furthermore, they provide a mapping between common known privacy-enhancing technologies to the identified privacy threats. This methodology is orthogonal to our methodology as well. In their methodology, for each system element, they provide a set of generic threats, and eventually, they suggest a set of privacy-enhancing technologies. However, they do not identify the specific privacy threats of a given system concerning its structure and behavior—how a piece of personal data is processed by the specific actions and behavior of the system.

Standards and legal methodologies to conduct *privacy impact assessment* and risk analysis. [59, 86, 159] provide methodologies and best practices to conduct a *PIA* in the UK, France, and Germany. In [196], the European Commission recommends a template to conduct a data protection impact assessment for smart grid and smart metering systems. Moreover, the ISO 27000 family of standards on information security management [119], and the ISO 31000 risk management standard [122] are recognized standards to keep information assets secure, and generally manage risks in organizations. In [60], ENISA provides a set of recommendations to assess the severity of personal data breaches. We leverage the principles and concepts of these legal methodologies, standards and recommendations to develop our *PIA* methodology, for instance, for creating the proposed control list, or performing a risk analysis. However, they are rather abstract and cannot be used as a concrete methodology to conduct a PIA.

Grimm et al. [93] provide a promising reference model to conduct an IT security analysis. Their reference model follows the IT baseline protection (IT Grundschutz) methodology [38]. They introduce a precise taxonomy of the terms related to the reference model such as the *world*, its *stakeholders*, and the *conflict of interests*. The IT security analysis is performed in four steps: **(I)** Analyzing the current condition of the system. **(II)** Identifying the threats and the security requirements at risk. **(III)** Establishing appropriate security measures, and **(IV)** integrating them into the system. The proposed concepts in this thesis, particularly our model-based *privacy impact assessment* methodology, can be integrated into this reference model to support an IT security analysis using system models. The *world* (current system condition) may be presented as a system model. Furthermore, the proposed privacy analysis and the UMLsec methodology may be used to identify the threats and the privacy and security targets at risk.

5.7 Preliminary Conclusion

We introduced a novel methodology to support *privacy impact assessment* using model-based privacy and security analyses. The methodology is based on BSI *PIA* and leverages two model-based privacy and security analyses to identify the system design violations and harmful activities. To fully support the privacy principles that are prescribed in the GDPR, we introduced three new privacy targets. Moreover, we presented a mechanism to calculate the impact of the threats on the privacy targets. We applied our methodology to industrial scenarios and provided a comparative evaluation with respect to three legal PIA methodologies.

In the following chapter, we introduce a methodology to enhance a system model with the suggested controls identified in a PIA.

Chapter 6

Privacy-Enhanced System Design Modeling Based on Privacy Features

This chapter shares material with the SAC'19 paper "Privacy-Enhanced System Design Modeling Based on Privacy Features" [10].

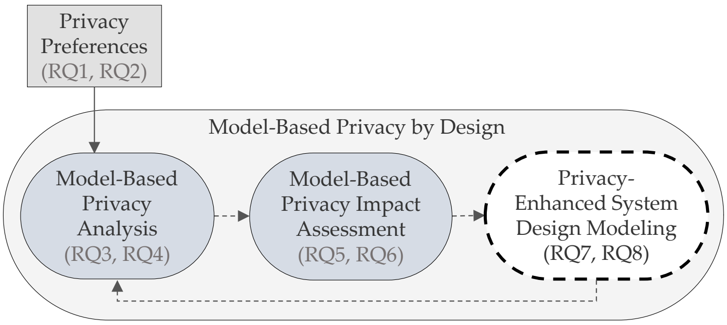

Privacy Preferences (RQ1, RQ2)
Model-Based Privacy by Design
Model-Based Privacy Analysis (RQ3, RQ4)
Model-Based Privacy Impact Assessment (RQ5, RQ6)
Privacy-Enhanced System Design Modeling (RQ7, RQ8)

Figure 6.1: The highlighted section (dashed lines) denotes how Chapter 6 contributes to the overall workflow (introduced in Section 2.3).

To ensure that their stakeholders' privacy concerns are addressed systematically from the early development phases (to operationalize *privacy by design*), organizations have to perform a privacy enhancement of the system design, in which appropriate technical and organizational controls are established. Such a privacy enhancement needs to account for three crucial types of input: First, risks to the rights of natural persons. Second, potential interrelations and dependencies among the privacy controls. Third, potential trade-offs regarding the costs of the controls. Despite numerous existing privacy-enhancing technologies and catalogs of privacy controls, there has been no systematic methodology to support privacy enhancement based on these types of input.

In this chapter, we propose a methodology to support the coherent privacy enhancement of a system model. We consider an extensive variety of privacy controls, including privacy-design strategies, patterns, and privacy-enhancing technologies. Representing these controls as privacy features, we explicitly maintain their interrelations and dependencies in a feature model [133]. In order to identify an adequate selection of controls, we leverage a model-based cost estimation approach that analyzes the associated costs and benefits. We further demonstrate how the selected features can be integrated into the system model, by applying reusable aspect models [136] to encapsulate the required changes to the system design.

6.1 Introduction

Article 25 of the GDPR [198] prescribes *privacy by design* (*PbD*), requiring service providers to implement appropriate technical and organizational controls from the early development phases for ensuring that the privacy concerns of their service customers and the privacy principles related to the processing personal data are addressed by design.

PbD mandates that the system design needs to be revised (enhanced) to incorporate the suggested controls, thus mitigating the risks. A privacy enhancement begins with a *privacy impact assessment* (*PIA*), which determines privacy threats by performing a systematic risk assessment and suggests potential technical and organizational controls to mitigate the privacy risks arising from those threats. For instance, to mitigate the risk of processing a piece of sensitive data for an unauthorized purpose, a control such as *data minimization* has to be integrated into the system. However, such privacy controls are too abstract to be integrated directly into the system design; instead, they may be established using one or multiple privacy-enhancing technologies (PETs), such as *anonymization* with *Mix Zones* [32].

Integrating an appropriate set of PETs into the system design is an intricate task that involves a number of sensitive aspects: **(I)** Some privacy risks are more pressing than others. Data owners have varying concerns about particular kinds of risks. For example, the leakage of email addresses may not be as problematic as that of national identification number or biometric data. According to the GDPR, the latter belong to special categories of personal data. **(II)** PETs can be related via various dependencies or conflicts. For example, the *authorization* to perform a particular task on data requires an *authentication*. **(III)** The implementation of PETs may come with various costs; implementing certain desirable PETs can be prohibitively expensive. Despite earlier work on security and privacy enhancement (discussed in Section 6.7), there is no methodology for improving an existing system design while simultaneously addressing these aspects.

In this chapter, we thus propose a systematic model-based methodology to coherently support the privacy enhancement of IT systems, addressing risks, interrelations, and costs in the above-mentioned sense. We use a set of privacy features that realize the privacy controls to conduct the enhancement. This methodology is based on system models expressed in UML [157].

As further input, our methodology takes the risks and controls identified while performing a *PIA*. We use our *PIA* methodology that we introduced in Chapter 5. Using the proposed methodology in this chapter, a privacy enhancement can be performed during the early stages of the system design. The result of the enhancement can be evaluated iteratively by experts by performing the *PIA* on enhanced system models. Specifically, we make the following contributions:

- We map the *NIST* privacy controls [154] to a set of privacy features, including privacy design strategies [52, 108], patterns [21, 52, 53, 80, 113, 160, 170, 183, 185], and privacy-enhancing technologies [36, 58, 73, 83, 201]. Furthermore, we identify conflicts and dependencies among these features and specify their interrelations using a feature model [133] (Section 6.4.1).

- To perform a cost-benefit analysis in our model-based privacy enhancement, we extend the cost estimation approach provided in [33] to make it applicable to reusable dataflow models (Section 6.4.2).

- To enable the integration of the features in the system design, first, we introduce a UML profile to establish traceability between privacy controls and model elements, and second, we propose to express the privacy enhancement by using and extending the concept of *Reusable Aspect Models* (*RAMs*) [136]. We extend RAMs with activity diagrams to specify *data flow* views, which are particularly important in our privacy setting (Section 6.4.3).

The remainder of this chapter is organized as follows. In Section 6.2, the necessary background is provided. In Section 6.3, we introduce an example and the research questions. In Section 6.4, we describe our methodology. In Section 6.5, we present our case studies. In Section 6.6, we discuss our results. In Section 6.7, we discuss related work. Finally, in Section 6.8, we conclude.

6.2 Background

Below, we present the necessary background for this chapter: We use *privacy design strategies* to support the transition from abstract *controls* to the use of concrete *privacy-enhancing technologies*, *function point analysis* to assess the cost for the selected controls and *reusable aspect models* to capture enhancements of the system design based on the selected controls.

6.2.1 Privacy Design Strategies, Patterns, and Privacy-Enhancing Technologies

A strategy describes a fundamental approach to achieve a certain goal. Hoepman [108] introduces eight privacy design strategies, which are derived from existing privacy principles and data protection laws, thus bridging the gap between the legal and the technical domain. To make the definition of these strategies more concrete, Colesky et al. [52] refine these eight strategies by defining a set of sub-strategies for each strategy and mapping each sub-strategy to a set of privacy patterns [160]. Originally, a pattern is more concrete than a strategy and describe a common recurring structure to solve a general design problem.

The term privacy-enhancing technology (PET) was originally introduced for a category of technologies with embedded privacy features that minimize the processing of personal data, and decrease the privacy risks for the user's data [58, 105]. PETs realize and implement privacy design patterns.

Privacy design strategies, patterns, and privacy-enhancing technologies may be affected by certain relationships and dependencies. For example, the *hide* strategy may not be applied together with the *inform* strategy. Such relationship represent necessary configuration knowledge for ensuring a valid use of the selected strategies. However, these relationships were not considered in the original systematization of privacy design strategies. In the present work, we analyze interactions between the considered strategies and formally specify the identified relationships using a feature model.

6.2.2 Function Point Analysis (FPA)

One of the essential prerequisites for successful software development is cost estimation. Most cost estimation models require to measure the functional size of a software to be developed [33]. The aim is to quantify the amount of functionality released to a user concerning the data that the software has to use to provide the functions, and the transactions through which the functionality is delivered. *Function Point Analysis* (*FPA*) [11] is one of the most commonly used functional size measurement methods. FPA identifies and weights data and transactional function types. Data functions represent data, and transactional functions represent operations that are relevant to the user. Data functions are classified into *internal logical files* (*ILF*), the data that is maintained within the boundary of an application, and *external interface files* (*EIF*), the data that is maintained outside the boundary of the application being measured. Transactional functions are classified into *external inputs* (*EI*), *external outputs* (*EO*), *external inquiries* (*EQ*). An EI processes an ILF. An EO presents data to a user. An EQ retrieves data from ILFs and EIFs. For every data or transactional function, different weights are defined.

In [33, 140], to estimate the cost of modeling, the authors apply FPA to UML models by defining a precise mapping between UML elements, and FPA's data and transactional functions. They focus on use-case, class, and sequence diagrams. In this chapter, we propose a mapping between activity diagram elements and FPA's data and transactional functions.

6.2.3 Reusable Aspect Models (RAMs)

*Reusable Aspect Model*s (*RAM*s) [136] is an aspect-oriented multi-view modeling approach for software design modeling. The paradigm of aspect orientation generally aims to identify, separate and represent crosscutting concerns. In RAM, the reusable concerns are modeled using UML *class* (structure view), *sequence* (message view), and *state* (state view) diagrams. A RAM may be (re)used within other models via its *usage* and *customization* interfaces. The former specifies the design structure and the behavior of the reusable model. The latter specifies how to adapt the reusable model using *parameterized* model elements (marked with a vertical bar |). A RAM model can be (re)used by composing the parameterized model elements with the elements of other models and RAMs. A RAM *weaver* is used to create a composed design model.

In [152], RAMs are used to model security patterns. We benefit from this work to perform the enhancement of a system model. However, in a privacy context, specifying and analyzing data flows in a system is crucial, which cannot be captured by the classical RAM diagram types—class, sequence, or state diagrams. In this chapter, we propose an extension of RAM based on activity diagrams to express the behavior (data flow) of a system.

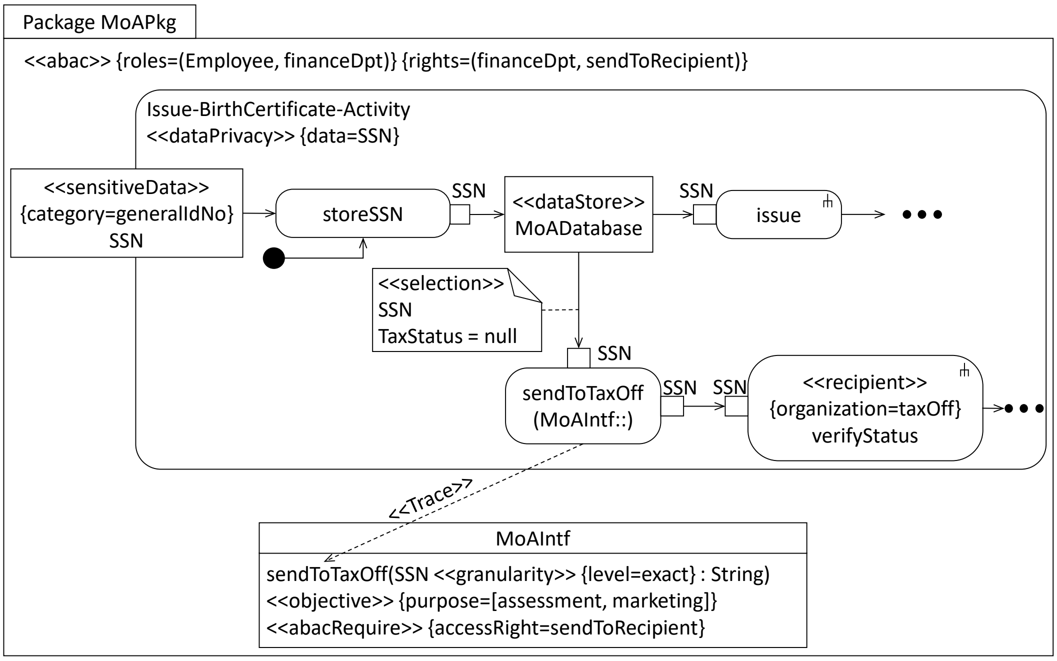

Package MoAPkg
<<abac>> {roles=(Employee, financeDpt)} {rights=(financeDpt, sendToRecipient)}
Issue-BirthCertificate-Activity
<<dataPrivacy>> {data=SSN}
<<sensitiveData>>
{category=generalIdNo}
SSN
storeSSN
SSN
<<dataStore>>
MoADatabase
SSN
issue
<<selection>>
SSN
TaxStatus = null
SSN
sendToTaxOff
(MoAIntf::)
SSN
SSN
<<recipient>>
{organization=taxOff}
verifyStatus
<<Trace>>
MoAIntf
sendToTaxOff(SSN <<granularity>> {level=exact} : String)
<<objective>> {purpose=[assessment, marketing]}
<<abacRequire>> {accessRight=sendToRecipient}

Figure 6.2: Design model excerpt (*issue birth certificate* activity).

6.3 Running Example

To describe the main problem and explain our privacy enhancement methodology, we refer to our example scenario introduced in Section 2.2 (*issuing a birth certificate* scenario). In Figure 6.2, we show once more an excerpt from the MoA (Municipality of the Athens) system model. The activity diagram expresses a business process in which the *SSN* (*Social Security Number*) is processed. In Section 3.3, we defined a set of privacy preferences for the *SSN*. Figure 6.3 shows the lattices, expressing the privacy preferences of the *SSN* (using dashed lines).

Performing our proposed *privacy impact assessment* methodology (Chapter 5) yields several privacy targets at risk. For instance, concerning the *sendToTaxOff* action and its corresponding operation in the class diagram, the *SSN* is processed for two purposes, namely *assessment*, and *marketing* (see «objective»). However, with respect to the lattices showed in Figure 6.3, the *SSN* should not be processed for the *marketing* purpose. Due to this privacy design violation, *P1.4 Ensuring limited processing for specified purposes* and *P5.2 facilitating the objection to direct marketing activities* are at risk.

The category of the *SSN* is *general ID number* ($PDCV = 1$). Assuming that both the data owner and the organization rate the impact value as *maximum* ($IV = 4$) (see

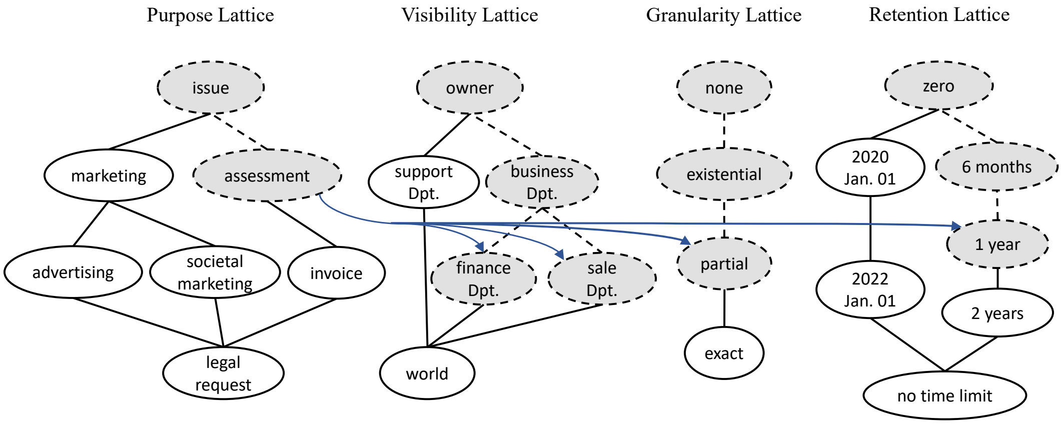


Figure 6.3: The privacy preferences of *SSN* (see Section 3.3). The dashed lines indicate the privacy preferences (authorized purposes, subjects, granularity levels, and retention conditions).

Table 5.4), the final impact score for both targets at risk are:

$$IA = 1 \times 4 \times 4 = 16$$

According to the ordinal scale provided in Table 5.5, a *very high* score. Following Section 5.3.5, to mitigate the risks, our *PIA* methodology suggests the following NIST privacy controls: For *P1.4*: ***AP-2*** *Purpose Specification* and **DM-1** *Minimization of Personally Identifiable Information*. For *P5.2*: **DM-1** and **TR-1** *Privacy Notice*.

The produced list of controls must be evaluated for applicability, a challenging task that involves two crucial questions:

RQ7: *How can an adequate selection of controls (concerning varying risks, interrelations between controls and the costs of controls) be identified to mitigate the identified privacy risks?*

We need to answer this question by taking into account the severity of the identified design violations (as captured by the impact score), possible interrelation and dependencies between controls and the costs for deploying the controls to the system.

RQ8: *How can the selected controls be incorporated into a system model?*

Following the privacy-by-design principle, we need to ensure that the system at hand is designed with the selected privacy controls in mind. To this end, the challenge is to enrich and expand the design model to account for the controls.

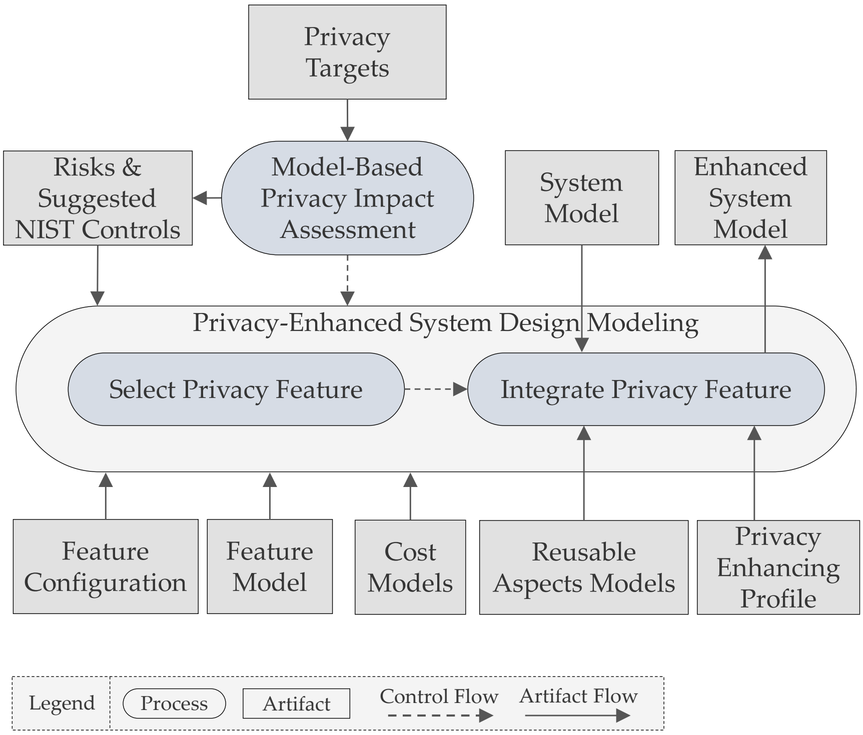


Figure 6.4: The workflow of the methodology proposed in this chapter to enhance a system model with privacy controls

6.4 Privacy-Enhanced System Design Modeling

Figure 6.4 provides an overview of our methodology to support the privacy enhancement considering risks, interrelations between the controls, and trade-offs regarding the costs of the controls. Our proposed *PIA* methodology (introduced in Chapter 5) is performed upfront to identify privacy risks and to suggest a list of NIST privacy controls to mitigate those risks. Our privacy enhancement of a system model is performed based on the suggested controls, a feature model of privacy design strategies, a cost model, and a set of reusable aspect models (RAMs).

First, we present a feature model (Section 6.4.1). A feature model captures the interrelations between privacy features—privacy design strategies and their refinement into patterns and PETs. Moreover, in a feature model, we introduce a mechanism to express the strength of privacy features to mitigate risks. To accomplish the main aim of an enhancement, which is the selection of proper privacy features, the relations between the privacy features and their strength are the most important factors. If there exist several features with the same strength to mitigate the risks, a cost estimation has to be conducted. Therefore, we introduce a model-based cost-estimation approach (Section 6.4.2). Finally, we show how the enhancement of a system model is performed using a UML profile and RAMs (Section 6.4.3).

6.4.1 The Privacy Design Strategies Feature Model

The purpose of the controls is to minimize, mitigate, or eliminate the identified privacy risks. Controls can be technical or non-technical; technical controls lend themselves to incorporation into the system. Nevertheless, the NIST technical controls are too generic to be directly integrated into a system model. For instance, ***DM-1*** *Minimization of Personally Identifiable Information* is a NIST privacy technical control. When integrating this control into the system model, one can rely on various data-minimization technologies and strategies, for example: exclude data from processing, define specific data processing purposes, or destroy data.

Hence, to apply the controls to system models, we map the NIST privacy controls to a set of *privacy design strategies*. Table 6.1 shows this mapping. As introduced in Section 6.2.1, we reuse a selection of eight privacy design strategies from existing work, including their concrete specifications using *sub-strategies*, *privacy design patterns* and *privacy-enhancing technologies* (PETs), thereby simplifying the realization of controls. The privacy design strategies, design patterns, and PETs provide an abstraction layer that enables the enhancement of system models with different levels of abstractions.

To map the design strategies to privacy design patterns—not included in Table 6.1—we leverage the correlations of the strategies and patterns which is provided in [52, 53]. We add a number of design patterns [21, 80, 113, 170, 183, 185] to refine this correlation. Eventually, we map each design pattern to one or more PET(s) [36, 58, 73, 83, 201]. The mapping between the NIST privacy controls and the privacy design strategies and the mapping between the privacy design patterns and the PETs, is achieved using an extensive literature review and argumentation. In Appendix F, we show the mappings between the design (sub-) strategies, patterns and PETs.

As a contribution of this work, we performed an investigation of interactions between the considered selection of privacy design-strategies, patterns, and PETs. To ensure that we can use them in our automated approach, we formalized the identified interactions using feature modeling. Feature modeling allows capturing variabilities in a system in terms of features and relationships between them. An excerpt of the resulting privacy-design-strategy feature model is presented in Figure 6.5. A feature model provides a tree-like hierarchy to structure different

Table 6.1: The mapping between privacy design strategies and the NIST privacy controls

Privacy Control (NIST)	**Design Strategy** [52, 108]
AP - Authority and Purpose	
(AP-1) Authority to Collect	Restrict
(AP-2) Purpose Specification	Consent
AR - Accountability, Audit, and Risk Management	
(AR-1) Governance and Privacy Program	Audit, Log, Report, Uphold
(AR-2) Privacy Impact and Risk Assessment	Report, Supply, Create
(AR-3) Privacy Requirements for Contractors and Service Providers	Demonstrate
(AR-4) Privacy Monitoring and Auditing	Demonstrate
(AR-5) Privacy Awareness and Training	Report, Supply, Explain
(AR-6) Privacy Reporting (AR-7) Privacy-Enhanced System Design and Development	Report All strategies
(AR-8) Accounting of Disclosures	Notify, Log, Report
DI - Data Quality and Integrity	
(DI-1) Data Quality	Update, Retract
(DI-2) Data Integrity and Data Integrity Board	Demonstrate
DM - Data Minimization and Retention	
(DM-1) Minimization of Personally Identifiable Information	Minimize, Hide
(DM-2) Data Retention and Disposal	Minimize, Hide
(DM-3) Minimization of PII Used in Testing, Training, and Research	Minimize
IP - Individual Participation and Redress	
(IP-1) Consent	Consent
(IP-2) Individual Access	Choose, Update, Retract
(IP-3) Redress	Update, Retract
(IP-4) Complaint Management	Demonstrate
SE - Security	
(SE-1) Inventory of Personally Identifiable Information	Supply, Update, Maintain, Uphold
(SE-2) Privacy Incident Response	Control, Enforce
TR - Transparency	
(TR-1) Privacy Notice	Notify
(TR-2) System of Records Notices and Privacy Act Statements	Demonstrate
(TR-3) Dissemination of Privacy Program Information	Demonstrate, Inform
UL - Use Limitation	
(UI-1) Internal Use	Minimize, Hide, Uphold
(UI-2) Information Sharing with Third Parties	Minimize, Demonstrate

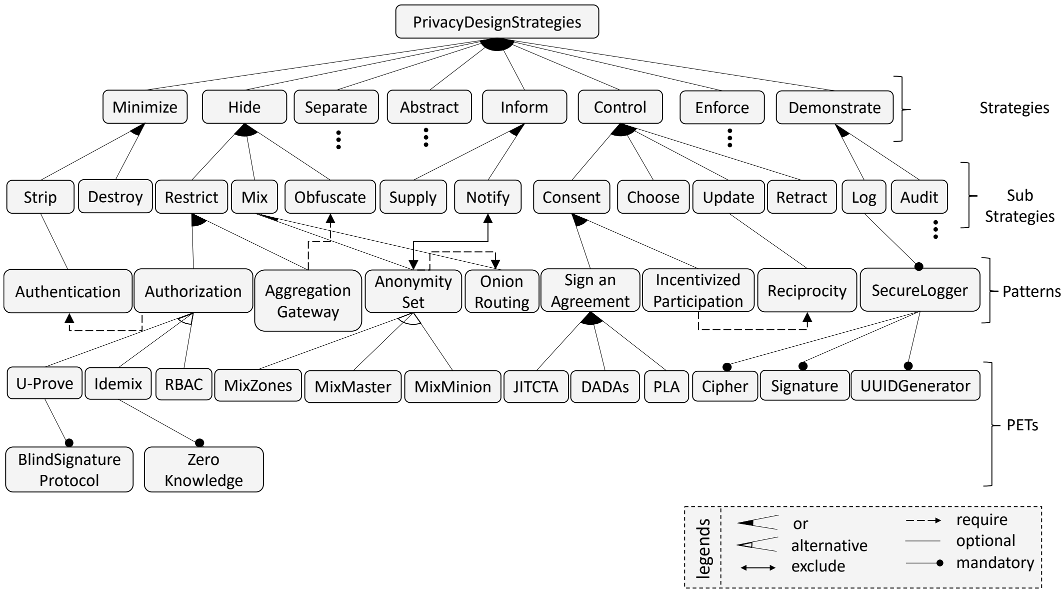


Figure 6.5: An excerpt of the feature model including privacy design strategies, sub-strategies, privacy patterns, and PETs

features. A child feature is either *mandatory* or *optional* for its parent feature. Furthermore, a feature model allows one to group a set of feature together with *or*-groups and *alternative*-groups. Where *or*-groups require at least one feature (from that group) be present if its parent feature is present, whereas *alternative*-groups require exactly one feature from that group to be present if its parent is present. Furthermore, a feature model allows us to define *require* and *exclude* relations between different features.

To investigate the interactions between the strategies, patterns, and PETs, we performed an extensive literature review. A few of these interactions are demonstrated in Figure 6.5 using the *require* and *exclude* relations. For instance, principally the anonymization is in conflict with transparency, therefore the strategies and patterns which use anonymization excludes the strategy related to transparency (for instance, *Inform* strategy) [58]. *Hierarchical attribute-based access control* [206] (a sub pattern of *Authorization*) requires encryption to provide the authorization mechanisms, therefore, it requires the strategy *Obfuscate* (not shown in Figure 6.5).

Furthermore, to enable an adequate selection of features to mitigate the privacy risks, similar to the privacy controls in Section 5.3.5, the features are classified based on the *levels of rigour*. The levels of rigour are defined as the attributes of the features and express the strength of the features to mitigate privacy risks with different severity levels. Similar to the classification of the controls in our *PIA* methodology (Section 5.3.5), the features are categorized into four levels of rigour, namely *suffi-*

cient, *medium*, *strong*, and *very strong*. The assignment of the levels of rigour to the features (as attribute) has to be performed before applying the methodology described in this chapter (for instance based on previous experiences or by a privacy expert).

In Figure 6.5, for the privacy enhancement of a system design model, initially, any of the eight privacy design strategies may be selected. For instance, if the strategy *Hide* is selected, optionally one or more sub-strategies might be chosen. If the sub-strategy *Restrict* is selected, one or more design pattern(s) may be selected. The design pattern *Authorization* may be realized by only one of the given technologies (*U-Prove*, *Idemix*, or *RBAC* (Role-Based Access Control)). According to the feature model, the technology *U-Prove* is further realized by the technology *Blind Signature Protocol*.

We created the feature model using the *FeatureIDE*[1] framework [134]. This framework also supports the configuration of features (that is, their assignment to *active* or *inactive*) based on a dedicated editor; configurations can be saved as configuration files. At the beginning, the configuration of features is an empty configuration of the feature model. In our work, we use a configuration to identify which features already exist or are modeled in a system model. The already existing features in a feature model are specified as active. We only show a representative excerpt of the feature model; the full feature model can be found online[2].

6.4.2 Model-Based Cost Estimation

To estimate the costs of privacy design strategies, patterns, and PETs, we propose a model-based cost estimation approach. The approach assumes that the design strategies, patterns, and PETs are modeled using activity diagrams, one of the main diagram types for specifying behavioral modeling in UML. *Functional point analysis* (*FPA*) has been used in [33, 140] to estimate costs in system models (use case, class, and sequence diagrams). As mentioned in Section 6.2.2, FPA aims at quantifying the amount of released functionality with respect to the data and the associated transactions used to supply the functions. In this section, we customize FPA for application to activity diagrams. This calls for a mapping between the elements of an activity diagram and *data function*s as well as *transactional function*s.

In an activity diagram, a piece of data is specified as an *ObjectNode*. An ObjectNode is fed into an action of an activity as a parameter. Concerning FPA, we identify an ObjectNode as a *data function*. We further need to classify an ObjectNode (*internal*

[1] `https://featureide.github.io/` (accessed: 2019-06-01)

[2] `https://cloud.uni-koblenz-landau.de/s/ocRXY9nJqDWzgpA` (accessed: 2019-06-01)

Table 6.2: Function types and their weights

Function type	Weight
ILF	7
EIF	5
EI	3
EO	4
EQ	3

logical file (*ILF*) or *external interface file EIF*). In the privacy profile introduced in Table 4.2, the stereotype «`recipient`» {`organization=value`} is introduced to annotate the actions (of an activity) that belong to a process outside the boundary of the process being analyzed. An ObjectNode that is fed into an action that is annotated with «`recipient`» is an EIF. All other ObjectNodes are ILFs.

An action in an activity diagram is a transactional function. An action which processes an ILF is an *external input* (*EI*). An action which processes an EIF is an *external output* (*EO*). An action which retrieves an object from a *DataStoreNode* (models a database in an activity diagram) with the UML selection behavior [157] (specified within a note symbol with the keyword «`selection`»), is an *EQ*.

Having mapped the FPA elements (functions) to UML activity's elements, to every function (either data or transaction) a complexity value, representing the number of FPs (function points) that the function contributes and a weight have to be assigned.

Following [33, 140], the complexity values can be acquired either on the basis of analogy (e.g., looking for previously measured data that contained similar data) or on the basis of the given system model. The weighting of function types may be obtained on the basis of their complexities. In [116] a precise general method to count FPs is provided. In our methodology, we use the approach presented by Bianco et al. [33] to assign complexity scores and weights to the functions. In this approach, the number of each function type in the model is multiplied by a predefined weight (Table 6.2) for the function type to calculate the final FP.

For instance, if in an activity diagram a which models a feature, two ILFs, one EIF, three EIs, one EO and one EQ are identified, then the total function point $FP(a)$ for the activity diagram a is:

$$FP(a) = (2 \times 7) + (1 \times 5) + (3 \times 3) + (1 \times 4) + (1 \times 3) = 35$$

If an activity diagram b with $FP(b) = 30$ exists, concerning the cost estimation approach provided in this section, we argue that activity a is more expensive than activity *b* (concerning the amount of functionality).

Table 6.2 only provides a baseline to calculate the FPs in an activity diagram. However, such weights can and should be modified (or classified), for instance by a deeper analogical analysis based on the experiences of the privacy experts and the feedback of the stakeholders, to obtain more precise weights.

6.4.3 Model-Based Privacy Enhancement

In our methodology, a design model specifies the structure and behavior of a system using class and activity diagrams. In a privacy enhancement, the selected privacy features are applied to design models, in particular, to their contained class and activity diagrams. The privacy enhancement is performed on two abstraction levels: **(I)** Establishing traceability between privacy features and affected design model elements via a dedicated profile. **(II)** Extending the structure and behavior with privacy features by applying *reusable aspect models* (*RAMs*).

In a nutshell, the privacy enhancement starts by identifying proper strategies concerning the suggested controls obtained by performing our proposed *PIA* upfront, and the mapping (Table 6.1) between the controls and the strategies. The selection of the proper sub-strategies, patterns and PETs (privacy features) takes into account the *impact score*s of the privacy targets at risks, and the *levels of rigour* of the features. Furthermore, the interrelations between the features specified by *require* and *exclude* relations in the feature model have to be considered. If a selection between two or more features from the same level of rigour is necessary, a cost analysis concerning the behavior of the features is performed to select a feature.

6.4.3.1 UML Profile for Privacy Enhancement.

To support the system developers in understanding which elements are affected by privacy concerns, we introduce a UML profile named *privacy-enhancing profile*. This profile can be used to automatically establish traceability between privacy features and model elements, by annotating the elements. Our profile includes one stereotype called «enhance» whose details we show in Table 6.3. A *Behavior* (an action in an activity diagram) may be annotated with «enhance» and its tags, namely *{strategy}*, *{pattern}*, and *{PET}* specifying the respective feature to be integrated into the action. The action has to be a *CallBehaviorAction* (indicated by placing a rake-style symbol), which calls a behavior including the behavior of the integrated feature.

The metamodel shown in Figure 6.6 demonstrates the underlying concept of the

Table 6.3: The privacy-enhancing profile with the «`enhance`» stereotype to express the privacy enhancement of a system model

Stereotype	Tags	UML Element	Description
«`enhance`»	strategy, pattern, PET	NamedElement	manifests the privacy enhancement of a NamedElement, for instance, an action (*CallBehaviorAction*) in an activity diagram, or a class in a class diagram.

enhancement. A privacy control is mapped to a set of (privacy) features which enhance a behavior in a system model. A behavior may have precondition and postcondition constraints. A constraint is an assertion that specifies a restriction that must be satisfied by any valid realization of the behavior containing the constraint [157]. We benefit from these constraints to regulate the control flow constraints specified by require and exclude relations. A feature f may require or exclude other feature(s). To verify whether a feature is required or excluded, using the preconditions' constraints, we investigate if the feature is *active* or *inactive* in the enhanced model element. The preconditions indicate the constraints that have to be held before invoking the feature f. In other words, the preconditions specify the features that have to be integrated into the system model (*active—require* relation) before integrating the feature f, and the features that must not exist in the system model (*inactive—exclude* relation) by integrating the feature f. Preconditions are evaluated on the given configuration of the feature model. Furthermore, the postcondition establishes a constraint that holds in the resulting system state, indicating that for instance, a feature (contained in the tag of «`enhance`») is integrated into the system model. The postconditions modify the configuration of a feature model and provide a basis to check the preconditions.

In our example (see Section 6.3), the *SSN* is processed for the unauthorized purpose *marketing*. As input for privacy enhancement, in Sections 5.3.4 and 5.3.5, we discussed how the privacy risks and a list of controls to mitigate those risks are determined in our *PIA*. Performing a *PIA* yields two privacy targets at risks, namely *P1.4* and *P5.2*. The final impact score for both is *very high*. The suggested controls to mitigate the risks arising from the privacy targets in danger, are **AP-2**, **DM-1**, **TR-1**. Concerning Table 6.1, these controls are mapped to four strategies: *Minimize, Hide, Notify* and *Consent*.

We show how a system model may be enhanced with the strategy *Hide*. Considering the feature model (Figure 6.5), *Hide* has three sub-strategies: *Restrict, Mix* and *Obfuscate*; the fourth one *Dissociate* is omitted for space reasons. We already mentioned that the features in Figure 6.5 are categorized based on four levels of rigour. Since the *impact score*s calculated for *P1.4* and *P5.2* are *very high*, only sub-strategies

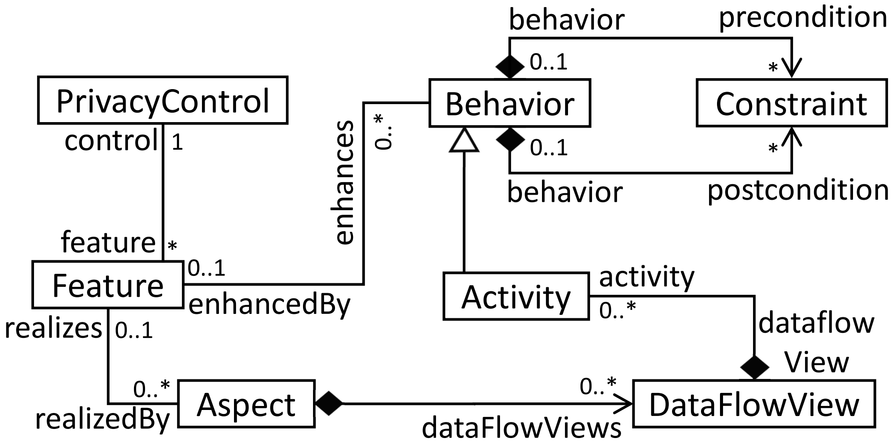


Figure 6.6: The metamodel demonstrating the underlying concepts of the privacy enhancement

with the rigour level *very strong* are considered, namely *Restrict* and *Mix*. With a similar argumentation, for sub-strategy *restrict*, the *Authorization* pattern may be considered and for sub-strategy *mix*, the *Anonymity Set*. Since the *Anonymity Set* excludes the *Notify* strategy and according to the initial set of mapped strategies, the *Notify* strategy has to be integrated into the system design, the sub-strategy *restrict*, and the *Authorization* pattern are used to enhance the model.

In Figure 6.7, the action *sendToTaxOff* is enhanced with the patterns that belong to the four strategies mentioned above. Since this action has to include the behavior of the patterns, it is demonstrated as a *CallBehaviorAction* (indicated by placing a rake-style symbol). Moreover, the *Authorization* pattern requires the *Authentication* pattern, this is expressed in the precondition constraint, and therefore, the action is not annotated with *Authentication*. In this activity diagram, the enhancement is performed using patterns, however, the enhancement can be applied by strategies or PETs.

An excerpt of the configuration of the feature model is provided in Figure 6.8. The postcondition constraints of an action (Figure 6.7) lead to the respective features in the configuration being active (the green + symbol) automatically. On activating the *Authorization* pattern, the *Authentication* pattern is activated as well, due to precondition constraints and *require* relation. Similarly, the activation of the *Notify* sub-strategy excludes the *AnonymitySet* pattern.

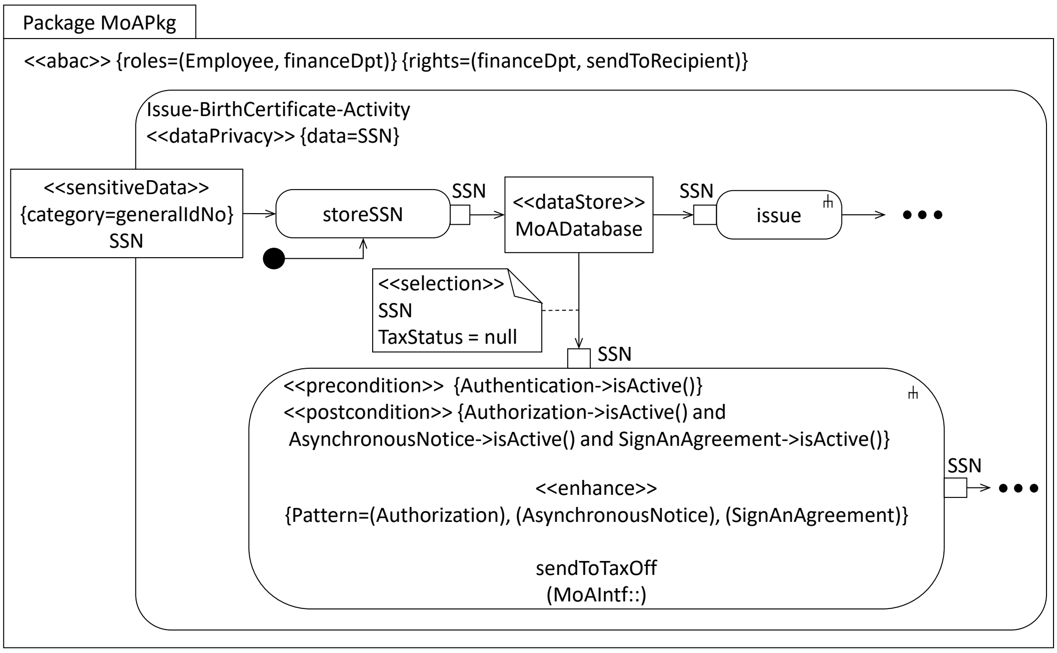
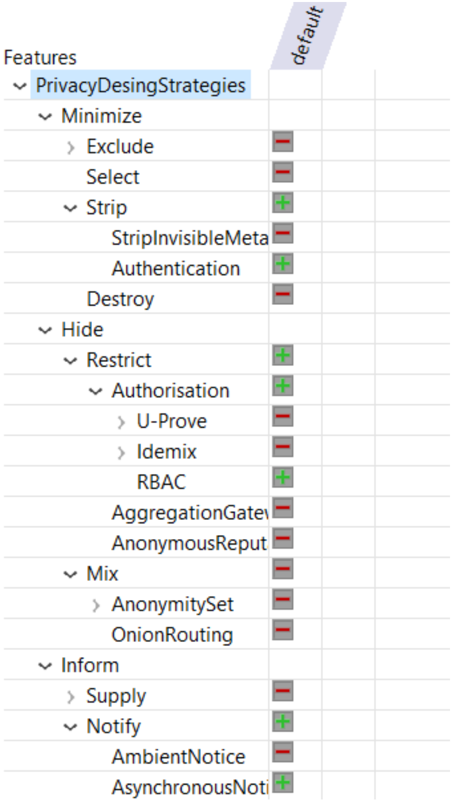

Package MoAPkg
<<abac>> {roles=(Employee, financeDpt)} {rights=(financeDpt, sendToRecipient)}
Issue-BirthCertificate-Activity
<<dataPrivacy>> {data=SSN}
<<sensitiveData>>
{category=generalIdNo}
SSN
storeSSN
SSN
<<dataStore>>
MoADatabase
SSN
issue
<<selection>>
SSN
TaxStatus = null
SSN
<<precondition>> {Authentication->isActive()}
<<postcondition>> {Authorization->isActive() and
AsynchronousNotice->isActive() and SignAnAgreement->isActive()}
<<enhance>>
{Pattern=(Authorization), (AsynchronousNotice), (SignAnAgreement)}
sendToTaxOff
(MoAIntf::)
SSN

Figure 6.7: An excerpt of the privacy-enhanced system model

6.4.3.2 Using and Extending RAMs for Privacy by Design.

In [152], the authors model security design patterns with *reusable aspect models* (*RAM*s) [136] to build a unified system of security design patterns that addresses multiple security concerns. While they do not consider privacy concerns and also focus on different diagram types than we do, we benefit from this work, since we can apply RAMs as well in order to encapsulate the required changes to the system model. For our privacy enhancement, we extend RAMs with a new kind of view called *data flow views*, which complement the existing structure and behavior views. As indicated in Figure 6.6, data flow views are modeled with activity diagrams, which are geared to capture privacy-relevant flows using object flows. Data flow views allow us to model the features as RAMs.

The activity diagram in Figure 6.7 is annotated with the *Authorization* pattern. This annotation specifies that the system model has to be revised by weaving the *Authorization* aspect into the system model. The *Authorization* aspect is demonstrated in Figure 6.9. The proposed *data flow view* in the provided RAM specifies that whenever a method is invoked on a protected class (pointcut), an authorization has to be performed before invoking the method (advice). If the access is granted (upon a successful evaluation of the request), the method will be invoked, otherwise an exception will be thrown (△ symbol). Since this aspect requires the *Authentication*



```
Features                                default
⌄ PrivacyDesingStrategies
  ⌄ Minimize
    › Exclude                           −
      Select                            −
    ⌄ Strip                             +
        StripInvisibleMeta              −
        Authentication                  +
      Destroy                           −
  ⌄ Hide
    ⌄ Restrict                          +
      ⌄ Authorisation                   +
        › U-Prove                       −
        › Idemix                        −
          RBAC                          +
        AggregationGate                 −
        AnonymousReput                  −
    ⌄ Mix                               −
      › AnonymitySet                    −
        OnionRouting                    −
  ⌄ Inform
    › Supply                            −
    ⌄ Notify                            +
        AmbientNotice                   −
        AsynchronousNoti                +
```

Figure 6.8: A screenshot showing an excerpt of a feature model's configuration in *FeatureIDE*

aspect, first the *Authentication* RAM has to be woven into the *Authorization* RAM. In [152], the *Authentication* aspect (without our proposed *data flow view*) and a description how to weave this aspect into *Authorization* RAM is provided. Moreover, handling the exception may be demonstrated in the *Authorization* aspect, or another RAM may be defined to handle such exceptions.

Similar to [136, 152], for weaving the aspects the generic weaver (GeKo) [147], a generic aspect-oriented model composition and weaving approach with available tool support, may be used. Furthermore, in [148], a formal specification for aspect weaving into activity diagrams is presented. This approach may be used to semantically apply a RAM weaver to activity diagrams.

As mentioned before, the process of applying a feature to a system model is based on the interrelations specified in the feature model, and the level of rigour identified by the final *privacy impact assessment* score. If two or more features from the same level of rigour are applicable to a system model, our model-based cost estimation approach from Section 6.4.2 identifies the appropriate feature. This approach is

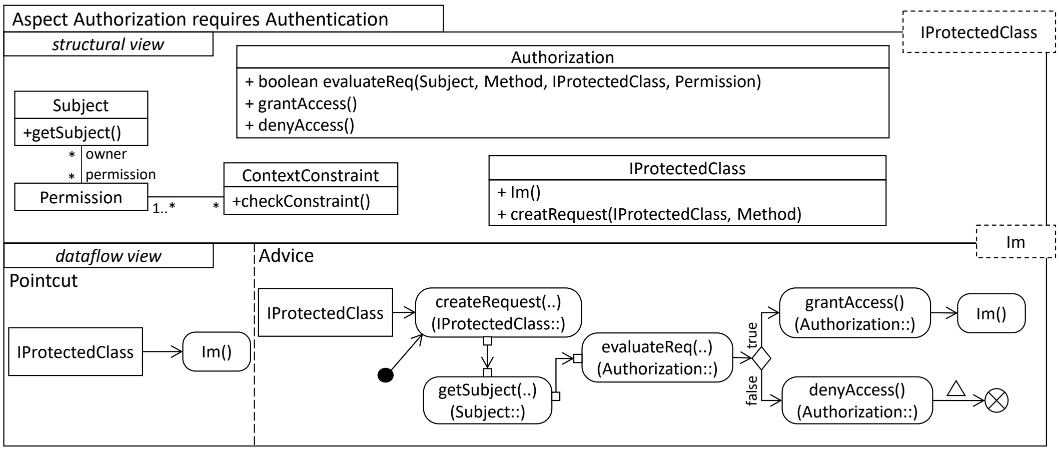


Figure 6.9: The *Authorization* aspect including a dataflow view

applicable to extended RAMs with *data flow views* as well. The activity diagram in the *Authorization* aspect has one ILF and four EIs. Since after the *DecisionNode* (◇), only one action is chosen, the number of EIs is four and not five. Based on Table 6.2, the final FP number for the *Authorization* aspect is:

$$19 = (1 \times 7) + (4 \times 3)$$

6.5 Case Studies and Evaluation

To evaluate the applicability of the privacy enhancement methodology proposed in this chapter, we applied it to three annotated system models provided by our industry partners (public administrations) in VisiOn project. We presented the details of these system models in Section 4.4.1. The system model which is analyzed in Section 6.3 and is enhanced in Section 6.4.3 is an excerpt of one of these system models. After performing our *PIA* in each case study, the system models are enhanced by applying the methodology introduced in this chapter.

The complete (*issuing birth certificate*) scenario (introduced in Section 6.3), eventually has been enhanced by seven design patterns. In Figure 6.7, we illustrated the privacy enhancement of this scenario. After performing a *PIA*, two more targets were at risk (besides the targets from the running example), namely *P1.8 Ensuring limited storage* and *P4.2 Facilitating the rectification, erasure or blocking of data* (Section 5.4). The privacy risks were the result of processing the *SSN* for the unauthorized *marketing* purpose and storing the *SSN* in a database (*DataStoreNode*) without implementing an appropriate mechanism to remove the *SSN* after it has been processed for the authorized purpose (*assessment*). To mitigate these risks, the system model has been enhanced by following controls: **DM-1**, **AP-2**, **TR-1**, **DM-2** and **IP-1** (strategies: *minimize*, *hide*, *consent* and *notify*).

In all three case studies, we identified the processing of sensitive data for unauthorized purposes. Therefore, the system models were enhanced by **DM-1** *Minimization of Personally Identifiable Information* and **AP-2** *Purpose Specification*.

Concerning the explored research questions:

RQ7: How can an adequate selection of controls (concerning varying risks, interrelations between controls, and the costs of controls) be identified to mitigate the identified privacy risks? Since the NIST controls are rather abstract, we mapped them to a set of features including strategies, patterns and PETs to mitigate the privacy risks identified by a privacy impact assessment. To capture the interrelations and dependencies between the features, we established a feature model. For each feature, a rigour level as an attribute is defined to mitigate the risks concerning the total impact scores (the severity of the violation in a system model). Furthermore, we applied *function point analysis* to system behavior models to enable a cost estimation of the features. For instance, concerning the data flow view of the *Authorization* aspect, we first identified the respective FPA elements and calculated the FPs for this aspect.

RQ8: How can the selected controls be incorporated into a system model? To support the privacy enhancement of the system models, our methodology provides a UML profile to annotate the system models. We further extended the reusable aspect models to encapsulate the behavior of the features expressed in the annotated models.

Below, we discuss the limitations of our proposed privacy-enhanced system design modeling methodology and provide the potential directions for future work.

6.6 Discussion and Limitations

Our approach requires a set of default (preexisting) values. For instance, the level of rigour of each feature has to be specified before applying the methodology. The levels of rigour have to be assigned by privacy experts. Learning from historical data, for instance, previous privacy enhancements may assist a privacy expert to refine the assignment of the levels of rigour to the privacy features. Moreover, to estimate the cost of the features, we used a set of predefined complexities for the data and transactional functions. In fact, using more rigorous complexities refines the estimations.

Our cost estimation approach relies on the assumption that effort can be estimated reliably in terms of element-counting metrics. In [67], a cost evaluation approach based on Butler's multi-attribute risk assessment framework [40] is introduced. In this approach, for each security control five different implementing costs are identified: installation cost (monetary), operation cost (monetary), system downtime (time), incompatibility cost (scale), and training cost (monetary). Prior to the cost evaluation, to each control c a cost x_i and respectively a weight w_i $(1 \leq i \leq 5)$ must be assigned. The total cost (TC) of a control c is calculated by

$$TC_c = \sum_{i=1}^{5} w_i V(x_i)$$

$V(x_i)$ is a value function that normalizes different unit measures (monetary, time, and scale) so that the values can be summed together. We may employ the similar approach, in which for each feature (strategy, pattern, or PET) five different implementing costs are defined, and eventually summed together. This is, in fact, helpful when an enhancement of a system model is only performed by applying the «*enhance*» stereotype, without weaving the RAMs. However, this kind of estimation requires that a privacy expert evaluates the different costs and weights, prior to the estimation.

In our evaluation, we only consider a limited number of models from three case studies, focusing on activity diagrams. In the future, to extend the evaluations of our proposed methodology, a larger set of cases with a larger selection of diagram types has to be studied.

As it is demonstrated in Figure 2.1, following a privacy enhancement, several iterations of the proposed methodology, including privacy analysis, privacy impact assessment, and privacy enhancement may be performed. To enable a privacy analysis in each iteration, we have to track the earlier enhancements in the previous iteration(s). To support this, we propose two different solutions as directions for future work. **(I)** In an enhancement, the feature configuration file is used to evaluate the preconditions. The feature configuration file may be extended to keep track of the enhancements in regard to each piece of data and the identified privacy design violations. **(II)** To perform an analysis in successive iterations, we may use a help report such as a table, determining the identified privacy design violations, and the corresponding privacy enhancement (in the previous iterations). In fact, such information is collected in a *PIA* report (see Section 5.3.6).

6.7 Related Work

Privacy-enhancing technologies (PETs) are not a new concept, and there exists a wide range of research on the PETs, privacy-patterns, controls, and design strategies. Moreover, there exist numerous methodologies and tools to support the privacy hardening and the selection of appropriate controls and PETs. We leverage related work and aims to reflect different aspects coherently in privacy hardening.

Nguyen et al. [152] present a model-based approach built on a system of security design patterns (*SoSPa*) to systematically automate the application of multiple security patterns in a system development. In this approach, the selection of the most appropriate features is only based on the interrelations between the patterns. The risks, their severities, and the privacy enhancement costs are not supported.

In [179], a promising approach to apply runtime reconfigurations to adaptive software systems using the concepts of product lines is provided. In [187], Soltani et al. propose a framework to employ a planning technique to automatically select suitable features that satisfy both the stakeholders' functional and non-functional requirements. These approaches neither precisely consider data privacy nor support the privacy enhancement of a system design in the early phases. Such approaches are orthogonal to our methodology. A research direction for future work is to investigate the integration of the system design and the run-time configurations.

Pearson et al. [170] propose a decision-based support system to assess contextual and environmental factors in product and service design. This decision-based support system is generic and does not consider the design of a system.

In [211] and [73], the authors provide different taxonomies and classifications of the PETs. In [73], a classification of the PETs is provided. In [102] a set of best practices on privacy hardening is provided. In [52, 108] a set of privacy design strategies are introduced. These works do not provide any mechanism to enhance a system design and select an appropriate strategy or a PET. However, they provide a conventional foundation to build the feature model introduced in this chapter.

Dewri et al. [67] provide a systematic approach to select a subset of security hardening controls concerning a trade-off between the overall cost of the projects and the security risks. In this approach, the design of a system is not considered, and not any mechanism to enhance a system after selecting the security hardening controls is provided.

In [194], Suphakul and Senivongse propose to use UML to model privacy design patterns based on the OECD (Organization for Economic Co-operation and Development) privacy principles [162]. They aim to help software developers to understand the prerequisites for ensuring privacy in a system design. Unlike reusable aspect models (RAMs), their patterns do not provide usage or customization interfaces. Furthermore, they do not describe how appropriate patterns can be identified to enhance a system design concerning different issues such as risks or the interrelation between the patterns. The concepts of their privacy design patterns can be added to the feature model proposed in this thesis.

In [19], Rivera et al. propose GuideMe, a 6-step systematic approach that assists the practitioners in eliciting a set of solution requirements from the GDPR principles—particularly the principles stated in Article 5. Solution requirements link the GDPR principles to a set of privacy controls necessary to satisfy them. In a nutshell, similar to the workflow of our model-based privacy by design methodology, GuideMe first determines where (in the flows, processes and systems) improvements are needed. In other words, first, the violations in regard to the GDPR principles are identified. Afterwards, several plans (requirement solutions) that determine what privacy controls are necessary for the corresponding improvements, are elicited. Our methodology is based on system models which provide an appropriate level of abstraction to capture the structure and the behavior of a system. In GuideMe, the authors claim that an analysis to identify the area of improvements may be performed by a model-based analysis, however such an analysis is not described. Moreover, they provide a privacy control catalog to support the requirement solution elicitation, however, an automatic (or semi-automatic) approach to identify proper controls that can be applied to the scenarios where violations occur and an improvement is necessary, does not exist.

Pullonen et al. [171] propose a set of privacy-enhanced extensions to the BPMN language for capturing data leakage in a business process. Using stereotypes, they provide a concrete syntax to enhance business processes with privacy-enhancing technologies. In this approach, the information flow analysis in the early phases of the system design is not supported.

6.8 Preliminary Conclusion

We have introduced a methodology for enhancing system models with privacy controls to mitigate privacy design violations during the design of a software system. Our enhancement methodology relies on our *PIA* methodology (introduced in Chapter 5) that identifies a set of risks and controls for mitigating the risks. Since the controls are rather abstract and cannot be directly integrated into the system design, we map them to more concrete privacy features, including strategies, design patterns and privacy-enhancing technologies (PETs). To determine an adequate selection of features, we take into account the severity of the identified design violations, possible interrelation and dependencies between the features, and the cost of integrating features into a system design. Furthermore, we performed an investigation of the interactions between the features and captured the respective interrelations and dependencies in a feature model. To estimate the cost of the strategies, patterns, and PETs, we proposed a model-based cost estimation approach by customizing functional point analysis for application to activity diagrams. Eventually, we introduced a UML profile to trace the privacy enhancement of a system model, and extended the concept of reusable aspect models to enhance a system behavior with appropriate privacy design strategies. We successfully applied our methodology to three case studies.

Chapter 7

Tool Support

This chapter shares material with the ESEC/FSE'17 paper "Model-Based Privacy and Security Analysis with CARiSMA" [6] and the BMSD'15 paper "Supporting the Security Certification and Privacy Level Agreements in the Context of Clouds" [5].

7.1 Introduction

In this chapter, we discuss elaborate tool support for the concepts presented earlier in this thesis. Our contribution is described within two research projects, namely, VisiOn[1] and ClouDAT[2]. In Chapters 2 and 4-6, we elaborated on VisiOn case studies that are used for the evaluation of our model-based *privacy by design* methodology. In this chapter, we describe the *VisiOn privacy platform* (*VPP*) that is developed during the VisiOn project to support the privacy of the EU citizens and assist the public administrations to consider privacy in their IT systems. We further introduce the *ClouDAT framework* that supports a security and privacy certification process by performing a risk assessment, suggesting appropriate controls to mitigate the identified risks and generating an ISO 27001 compliant documentation based on the outcomings of the risk assessments.

Our concepts introduced in the previous chapters are mainly integrated into the CARiSMA tool[3]. CARiSMA is originally developed to implement UMLsec checks

[1] `http://www.cloudat.de/` (accessed: 2019-06-01)

[2] `http://www.visioneuproject.eu/` (accessed: 2019-06-01)

[3] `http://carisma.umlsec.de` (accessed: 2019-06-01)

and conduct security analysis on UML diagrams. In this thesis, CARiSMA enables one (a system developer, a practitioner in a public administration or privacy analyst) to conduct a privacy analysis on a UML system model and supports the identification of privacy risks.

The remainder of this chapter is organized as follows. In Section 7.2, we introduce the VPP and elaborate on our contribution to extend CARiSMA to support our proposed privacy analysis. In Section 7.3, we introduce the risk analysis process provided in the ClouDAT framework and explain how this framework supports a privacy impact assessment.

7.2 Model-Based Privacy Analysis with CARiSMA in the Context of the VisiOn Privacy Platform

Nowadays, IT service providers increasingly require personal data of their customers to perform their services [190]. For instance, public administrations such as hospitals or administration offices of municipalities are offering more and more IT services to patients and citizens. Such services enormously involve personal data processing. Although these services have many benefits, new security and privacy risks emerge, when security and privacy concerns are not appropriately supported during the development process [54].

In this thesis, we introduced a model-based methodology to operationalize *Privacy by design* (*PbD*). *PbD* implies that appropriate controls must be integrated into a system design from early phases of the system development. Concerning the challenges that we identified in Section 1.1, operationalizing (*PbD*) calls for a system model analysis to identify privacy design violations.

CARiSMA has been designed to support the security analysis of IT systems in a model-based manner using the UML extension UMLsec [128] by providing a set of security checks. Two security checks of CARiSMA, namely, *secure links* and *secure dependency*, are introduced in Sections 4.2.3.1 and 4.2.3.2. In Section 5.3.4, we showed that the UMLsec checks together with our proposed privacy checks are used to conduct a privacy impact assessment (see Tables 5.1 and 5.2). CARiSMA originally does not enable a system developer to express privacy concerns in a system's design or perform a privacy analysis. Therefore, we extended CARiSMA to support the concepts provided in Chapters 3 and 4.

CARiSMA enables system developers to express security requirements [91] such as confidentiality, integrity, and availability within system models using UMLsec profile. Concerning our privacy extension (which will be introduced in this section), it further enables one to express the privacy issues in a system model. To perform an analysis, CARiSMA requires annotated UML system models. However, annotating the system models properly and initializing appropriate CARiSMA analysis are challenging tasks. A CARiSMA analysis includes a set of privacy and security checks to analyze a system model. CARiSMA provides no automatic mechanism to assist system developers in performing an analysis concerning given security requirements and privacy concerns. In other words, a system developer has to manually analyze the requirements and perform an appropriate analysis. In Section 4.4.2.2, we denoted that according to our industry partners' reports, annotating the system models with the privacy and security profiles, required more efforts than modeling the PA systems.

After performing a security analysis, CARiSMA provides a set of analysis results. The analysis results may contain information on design violations in a system model. In Section 5.3.4, we mentioned that such violations have to be further evaluated to identify privacy risks and conduct an impact assessment. CARiSMA does not provide additional tool support for automated or assisted evaluation of the analysis results.

Based on these considerations, we introduce the following new functionalities:

- **Analyzing security and privacy requirements** to automatically initialize analyses and assist system developers with annotating the system models.
- **Role-attribute-based access control** to support model-based privacy analysis of system models.
- **Evaluating analysis results** to generate appropriate questions to collect feedback on potential conflicts between system's design, and security and privacy requirements of citizens.

Concerning a set of privacy preferences and security requirements, we explain how a developer may be assisted to express the privacy and security issues within a system model, and how automatically a system model may be analyzed (Sections 7.2.1 and 7.2.2). The functionalities mentioned above are added to the CARiSMA during the VisiOn project and support the integration of CARiSMA into the VPP.

In Chapter 3 we introduced the concepts of privacy preferences and privacy level agreements (PLAs). The VPP, is further used to establish agreements on the use of personal data between citizens and public administrations (besides performing various privacy and security analysis). We explain the use of VisiOn PLAs within the VPP which supports our definition of PLAs (Section 7.2.3). Several tools that are

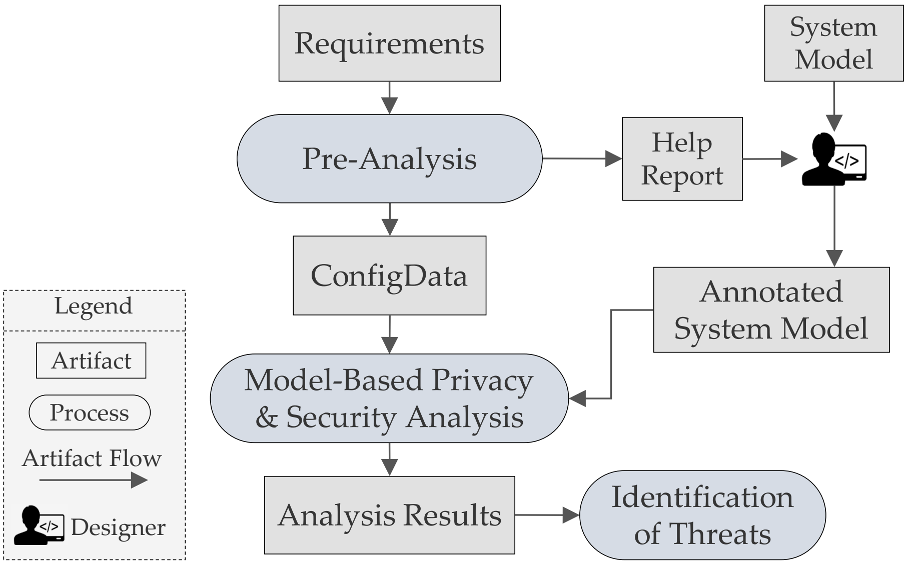


Figure 7.1: Model-based security and privacy analysis

integrated into the VPP (including CARiSMA) benefit from the PLAs to perform various types of analysis (for instance, privacy requirements analysis and threat analysis).

7.2.1 Overview and New Features

Figure 7.1 demonstrates how the workflow of performing a privacy analysis (showed in Figures 4.4 and 5.3) is extended to assist a system developer with annotating a system model and to automatically initialize an analysis with respect to a set of privacy and security requirements. Given a UML system model as well as privacy and security requirements, first, a system developer performs a pre-analysis. The results of this pre-analysis are:

- **A help report** that assists a system developer to express the security and privacy requirements within system models. Using the help report, a system developer annotates a system model with the security and privacy requirements, and eventually, runs a CARiSMA analysis to analyze the system model.
- **Configuration data** that automatically initializes a CARiSMA analysis concerning the given requirements.

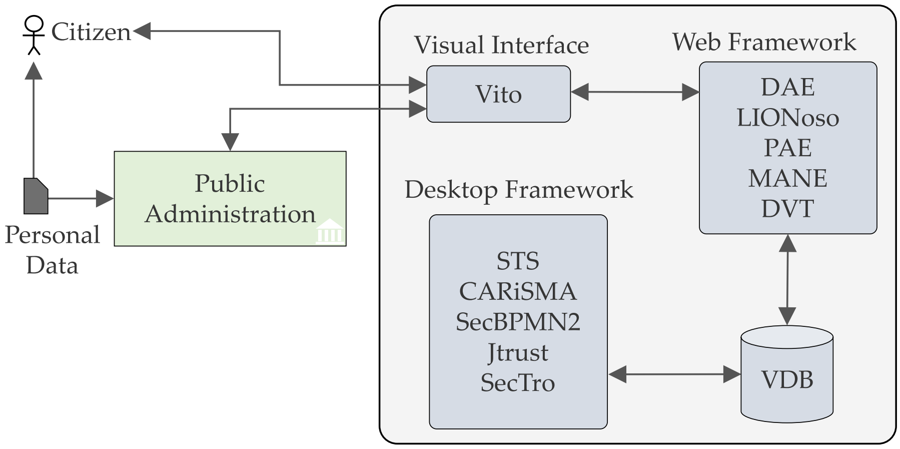


Figure 7.2: An excerpt from the architecture of the VisiOn privacy platform (VPP).

The analysis is based on CARiSMA's security checks (UMLsec checks [128]) and our proposed privacy checks (introduced in Chapter 4). The analysis results of such checks can be further evaluated afterwards, for instance, for the generation of privacy-related questions. Such questions are used to perform a *privacy impact assessment* (Chapter 5). In the following section, in the context of the VisiOn project, we explain the above-mentioned features added to CARiSMA.

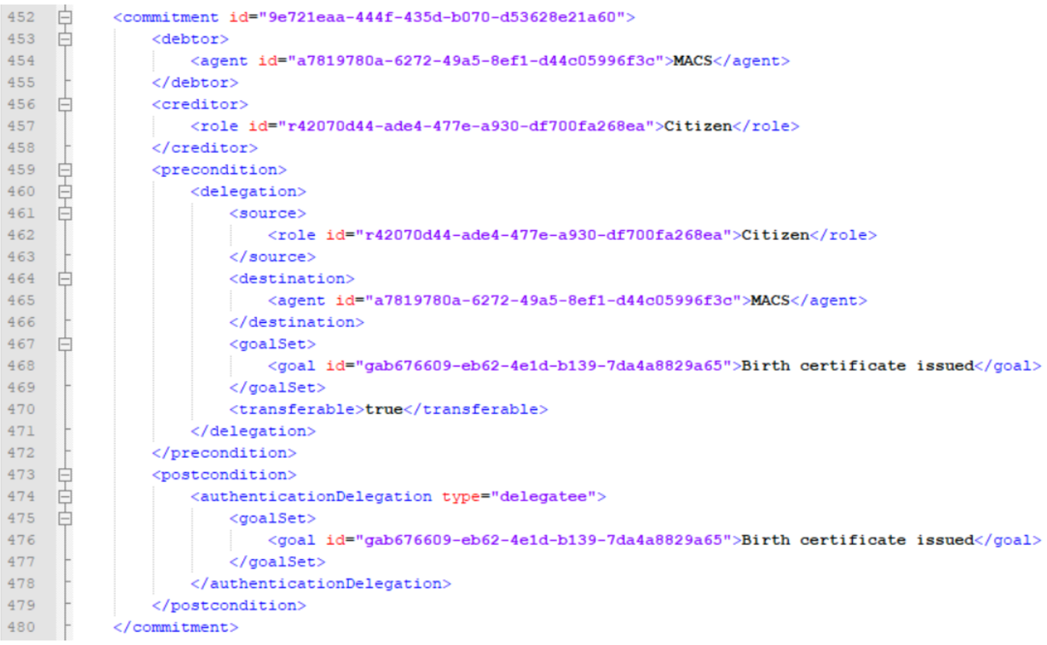
7.2.2 Security and Privacy Analysis within the VisiOn Privacy Platform

This section mainly describes the workflow of performing privacy and security analyses with CARiSMA within the VPP. We first, explain the architecture of the VPP. Afterwards, we explain the integration of CARiSMA into the VPP, and demonstrate the process of performing an analysis on a system model.

7.2.2.1 The Architecture of VPP

In the context of the VisiOn project, the VPP for evaluating and analyzing privacy levels of a *public administration* (PA) system is developed. This platform is further used to establish agreements on the use of personal data between a citizen and the PAs, and between each two PAs, to enforce privacy policies.

The architecture of the VPP is demonstrated in Figure 7.2. It is composed of four components:



```xml
<commitment id="9e721eaa-444f-435d-b070-d53628e21a60">
    <debtor>
        <agent id="a7819780a-6272-49a5-8ef1-d44c05996f3c">MACS</agent>
    </debtor>
    <creditor>
        <role id="r42070d44-ade4-477e-a930-df700fa268ea">Citizen</role>
    </creditor>
    <precondition>
        <delegation>
            <source>
                <role id="r42070d44-ade4-477e-a930-df700fa268ea">Citizen</role>
            </source>
            <destination>
                <agent id="a7819780a-6272-49a5-8ef1-d44c05996f3c">MACS</agent>
            </destination>
            <goalSet>
                <goal id="gab676609-eb62-4e1d-b139-7da4a8829a65">Birth certificate issued</goal>
            </goalSet>
            <transferable>true</transferable>
        </delegation>
    </precondition>
    <postcondition>
        <authenticationDelegation type="delegatee">
            <goalSet>
                <goal id="gab676609-eb62-4e1d-b139-7da4a8829a65">Birth certificate issued</goal>
            </goalSet>
        </authenticationDelegation>
    </postcondition>
</commitment>
```

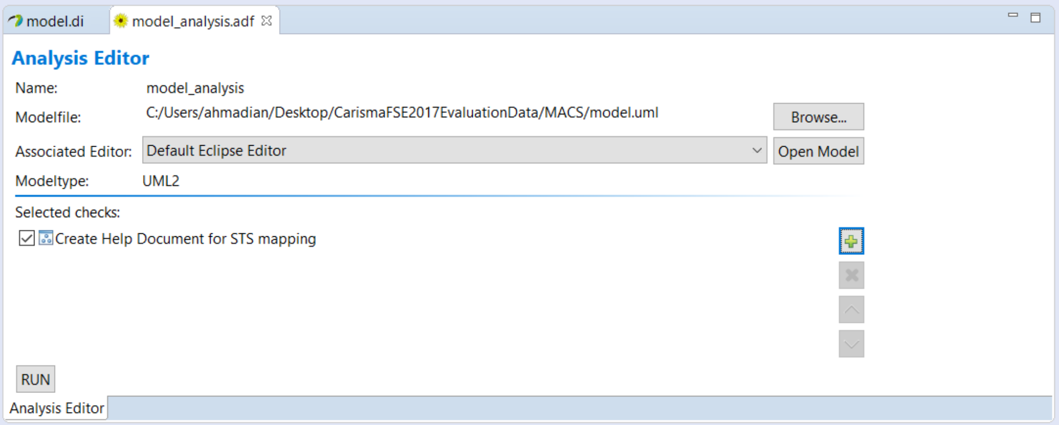
Figure 7.3: A document excerpt generated by STS, listing the security and privacy requirements

- The visual interface to a PA is realized by the *Vito* tool.
- The *VisiOn Database* (*VDB*) stores all models, agreements metadata and analysis results.
- The *web framework* gathers the feedback, preferences, and requirements of the PAs and enforces the agreements.
- The *desktop framework* includes different modeling tools to model the requirements and performs privacy as well as security analysis.

CARiSMA is integrated into the *desktop framework*, and it verifies whether a PA system supports the privacy preferences and the security requirements derived originally from the citizen's privacy preferences and the legal requirements.

Initially, the VPP provides a set of questionnaires (through the *web framework*) to elicit the privacy preferences and the security requirements. The results of these questionnaires are modeled with the requirement modeling tool *STS*[4] [164, 165]. STS models are stored in the VDB and may be transferred to other tools in the VPP such as CARiSMA for further analysis. In the following section, using our

[4] `http://www.sts-tool.eu/` (accessed: 2019-06-01)



Analysis Editor

Name: model_analysis

Modelfile: C:/Users/ahmadian/Desktop/CarismaFSE2017EvaluationData/MACS/model.uml

Associated Editor: Default Eclipse Editor

Modeltype: UML2

Selected checks:

☑ Create Help Document for STS mapping

Figure 7.4: A screenshot excerpt of CARiSMA demonstrating the pre-analysis. A STS model specifying the security requirements and the privacy preferences may be read from a local file or automatically from the VisiOn database (VDB).

example scenario (introduced in Section 2), we demonstrate how a privacy analysis (including a pre-analysis) is conducted by CARiSMA relying on the STS models.

7.2.2.2 System Model Analysis Using CARiSMA

This section refers to our example scenario derived from the MoA's case study. In Section 4.4.1, we introduced the three case studies of the VisiOn project. MoA is a public administration.

The MoA's system model (*issuing a birth certificate*) is either already modeled by the system developer or is available as a part of the system specification. In a system model analysis, first, using CARiSMA, a pre-analysis is performed on the MoA's system model. This pre-analysis facilitates the annotation of a system model with the privacy and security profiles and initializes a CARiSMA analysis. To perform a pre-analysis, CARiSMA provides an option to automatically read STS models from the VisiOn database or a local file. STS models specify the security requirements and the privacy preferences. STS models are stored as XML files in the VisiOn database. An excerpt of such an XML file is shown in Figure 7.3. The *Create Help Document for STS Mapping* check (Figure 7.4) performs a pre-analysis.

After running the pre-analysis, the CARiSMA's *results view* offers different options to handle the check results, showing an excerpt in Figure 7.5. A textual *help report* may be created, which assists a system developer or a PA administrator with annotating the system models. Figure 7.6 shows an example of such a report. The commitment in Figure 7.6 relates to the commitment (requirement) listed in Fig-

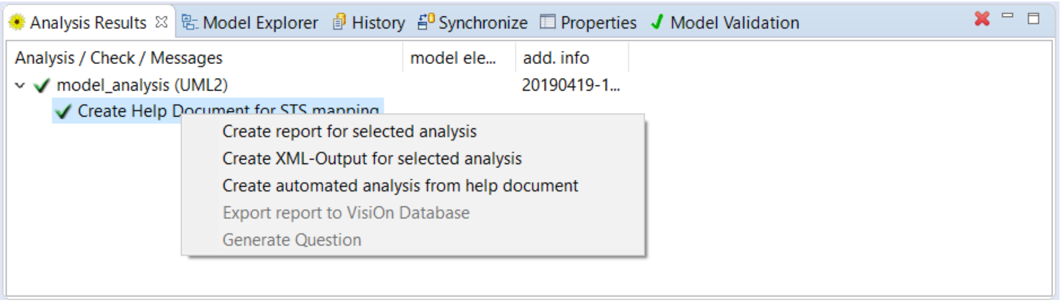
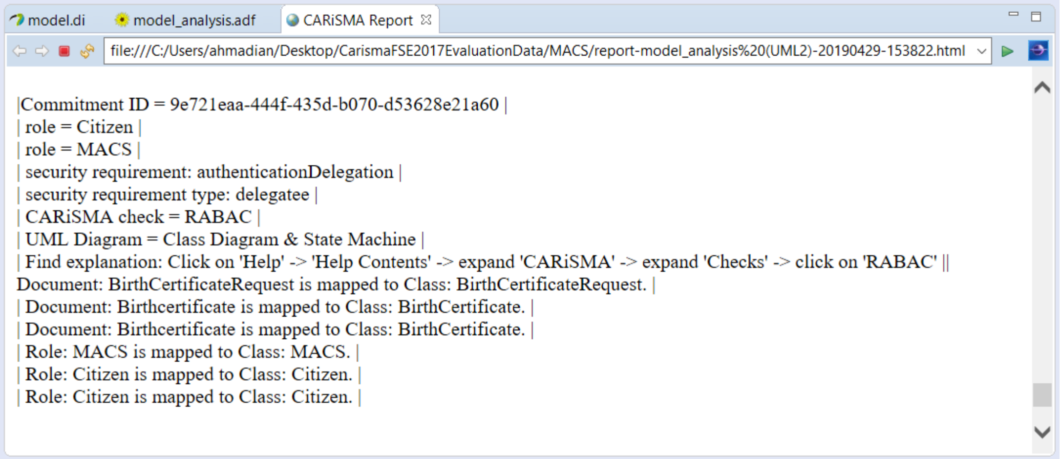

Analysis Results
Model Explorer
History
Synchronize
Properties
Model Validation
Analysis / Check / Messages
model ele...
add. info
model_analysis (UML2)
20190419-1...
Create Help Document for STS mapping
Create report for selected analysis
Create XML-Output for selected analysis
Create automated analysis from help document
Export report to VisiOn Database
Generate Question

Figure 7.5: CARiSMA offers several options after running a pre-analysis.

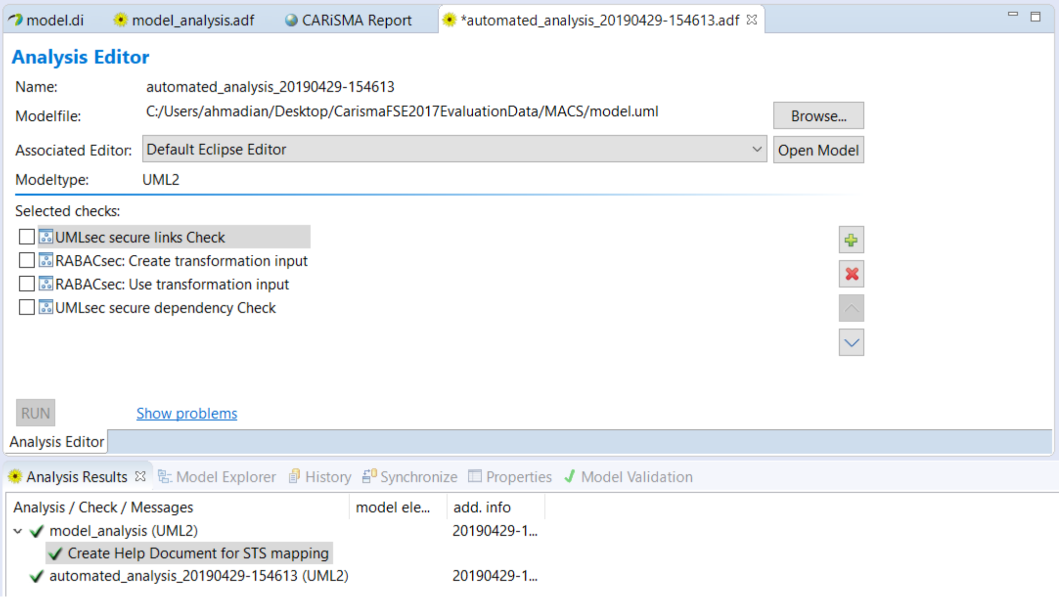

model.di
model_analysis.adf
CARiSMA Report
file:///C:/Users/ahmadian/Desktop/CarismaFSE2017EvaluationData/MACS/report-model_analysis%20(UML2)-20190429-153822.html

```
|Commitment ID = 9e721eaa-444f-435d-b070-d53628e21a60 |
| role = Citizen |
| role = MACS |
| security requirement: authenticationDelegation |
| security requirement type: delegatee |
| CARiSMA check = RABAC |
| UML Diagram = Class Diagram & State Machine |
| Find explanation: Click on 'Help' -> 'Help Contents' -> expand 'CARiSMA' -> expand 'Checks' -> click on 'RABAC' ||
Document: BirthCertificateRequest is mapped to Class: BirthCertificateRequest. |
| Document: Birthcertificate is mapped to Class: BirthCertificate. |
| Document: Birthcertificate is mapped to Class: BirthCertificate. |
| Role: MACS is mapped to Class: MACS. |
| Role: Citizen is mapped to Class: Citizen. |
| Role: Citizen is mapped to Class: Citizen. |
```

Figure 7.6: A screenshot excerpt of CARiSMA showing a help report excerpt generated by CARiSMA after running the pre-analysis

ure 7.3. This report specifies which roles (agents) and security requirements are involved in the STS model. Furthermore, it is specified which CARiSMA check may be used to perform an analysis and which models are required. The RABAC check has to be performed. Finally, a mapping between the STS model elements and the UML system model elements is provided. This mapping, in fact, shows which elements of a system model have to be annotated. A system developer or a PA administrator leverages the produced help report to apply appropriate UML profiles (e.g., the UMLsec or the privacy profile) and the stereotypes defined in these profiles to corresponding model elements.

Moreover, out of the results of a pre-analysis, a CARiSMA analysis that contains security and privacy checks may be automatically generated. A CARiSMA analysis is used to analyze the system model. For instance, Figure 7.7 demonstrates an automatically generated CARiSMA analysis, which indicates that various checks (two UMLsec checks and the RABAC check) have to be performed to analyze a system model. In the following section, we introduce the RABAC check.



Analysis Editor

Name: automated_analysis_20190429-154613

Modelfile: C:/Users/ahmadian/Desktop/CarismaFSE2017EvaluationData/MACS/model.uml

Associated Editor: Default Eclipse Editor

Modeltype: UML2

Selected checks:

- ☐ UMLsec secure links Check
- ☐ RABACsec: Create transformation input
- ☐ RABACsec: Use transformation input
- ☐ UMLsec secure dependency Check

Analysis / Check / Messages	model ele...	add. info
✓ model_analysis (UML2)		20190429-1...
✓ Create Help Document for STS mapping		
✓ automated_analysis_20190429-154613 (UML2)		20190429-1...

Figure 7.7: A screenshot excerpt of CARiSMA showing an automatically generated CARiSMA analysis after running a pre-analysis

7.2.2.3 RABAC

RABAC (*role-attribute-based access control*) is a CARiSMA plugin to analyze the access to protected items (such as a piece of personal data, or an operation that processes a piece of personal data) in UML system models. It further supports the VPP to perform privacy and security checks. The RABAC check provides prototypical tool support to perform a privacy analysis (particularly the *visibility* check) which is introduced in Section 4.3.4. In this section, we explain the details of the RABAC[5].

In Section 4.3.3, we briefly introduced the *rabac* profile. To perform a RABAC check, first, a system model has to be annotated with the *rabac* profile. The *rabac* profile includes three stereotypes: «`abac`», «`abacAttribute`», «`abacRequire`». They are listed in Table 7.1.

The stereotype «`abac`» specifies the basis of the access control. It contains the tags:

- The tag *roles* defines different roles for the subjects (denoted as a 2-tuple).
- The tag *rights* assigns different rights to the roles (denoted as a 2-tuple).
- The tag *rh* defines a partially ordered set over roles specifying inheritance

[5] RABAC is implemented within a Bachelor Thesis [109] supervised by the author of this thesis.

Table 7.1: *rabac* profile

Stereotype	Tag	UML Element	Description
«abac»	roles rights rh ssd dsd attributeFilters	Package	enforces role-attribute-based access control
«abacAttribute»	name	Operation	rabac for an attribute
«abacRequire»	accessRight filters	Operation	rabac for an operation

relations between the roles. Using an inheritance relation, a role inherits the rights assigned to another role. This tag is not relevant for a privacy or a security analysis in this thesis.

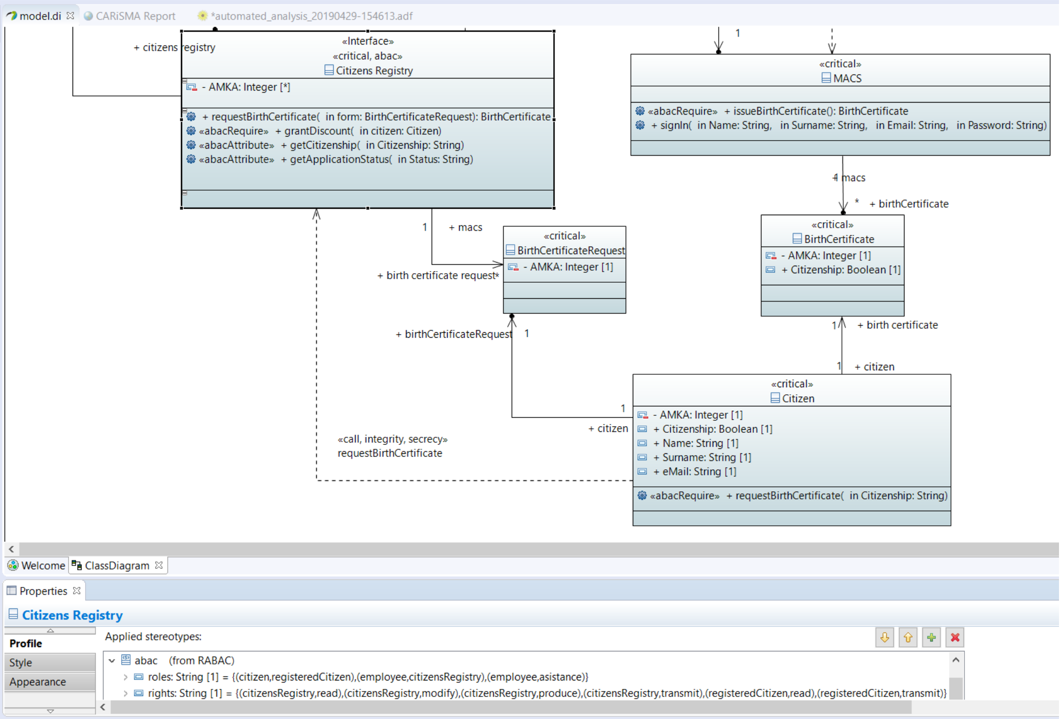
- The tag *attributeFilters* is used for filters which will be used to allow or deny rights. A filter is written in Object Constraint Language (OCL) and can handle the keywords *and*, *or*, *exists* and *forAll*. The filters are used globally for all access control analyses with respect to the attributes.

The stereotype «abacAttribute» annotates the operations that return an attribute. The attributes which are used in the *attributeFilter* have to be similar to the returning values.

The stereotype «abacRequire» annotates operations as well. It defines the rights that are required to execute an operation. Furthermore, the tag *filters* assigns a set of attribute and their corresponding values to an operation. The filters are defined similarly to the «abac» filters. The «abacRequire» stereotype may be used further to annotate the transitions of a UML state diagram as well [109]. In this thesis, the «abacRequire» stereotype is only used to annotate the operations in a class diagram.

In the context of the *visibility* check (Section 4.3.4), only the two stereotypes «abac» and «abacRequire» are used to annotate the models and perform a privacy analysis to verify who has access to a piece of personal data. Using the tag *attributeFilters* and the stereotype «abacAttribute», the RABAC check further verifies the rights to access a piece of personal data concerning a set of specific attributes.

To describe the RABAC check, we use the system model of the MoA to demonstrate the check. Before performing the check, the system model (the class diagram) has to be annotated with the above-mentioned stereotypes and the corresponding tags.


model.di
CARiSMA Report
*automated_analysis_20190429-154613.adf
+ citizens registry
«Interface»
«critical, abac»
Citizens Registry
- AMKA: Integer [*]
+ requestBirthCertificate(in form: BirthCertificateRequest): BirthCertificate
«abacRequire» + grantDiscount(in citizen: Citizen)
«abacAttribute» + getCitizenship(in Citizenship: String)
«abacAttribute» + getApplicationStatus(in Status: String)
1
«critical»
MACS
«abacRequire» + issueBirthCertificate(): BirthCertificate
+ signIn(in Name: String, in Surname: String, in Email: String, in Password: String)
1
+ macs
«critical»
BirthCertificateRequest
- AMKA: Integer [1]
+ birth certificate request*
4 macs
* + birthCertificate
«critical»
BirthCertificate
- AMKA: Integer [1]
+ Citizenship: Boolean [1]
+ birthCertificateRequest 1
1 + birth certificate
1 + citizen
1
+ citizen
«call, integrity, secrecy»
requestBirthCertificate
«critical»
Citizen
- AMKA: Integer [1]
+ Citizenship: Boolean [1]
+ Name: String [1]
+ Surname: String [1]
+ eMail: String [1]
«abacRequire» + requestBirthCertificate(in Citizenship: String)
Welcome
ClassDiagram
Properties
Profile
Style
Appearance

Citizens Registry

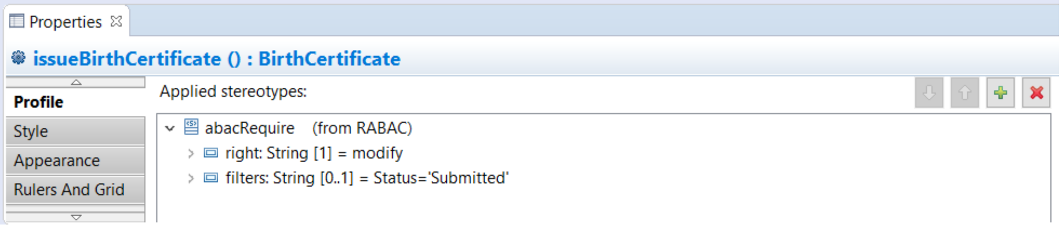
Applied stereotypes:

abac (from RABAC)
roles: String [1] = {(citizen,registeredCitizen),(employee,citizensRegistry),(employee,asistance)}
rights: String [1] = {(citizensRegistry,read),(citizensRegistry,modify),(citizensRegistry,produce),(citizensRegistry,transmit),(registeredCitizen,read),(registeredCitizen,transmit)}

Figure 7.8: A class diagram excerpt of the MoA system model

Figure 7.8 demonstrates a class diagram excerpt of the MoA system model. The RABAC analysis relies on the definition of the roles, rights, and attributes. In Figure 7.8, the roles and the rights are specified in the *Citizen Registry* class, using the «`abac`» stereotype. The properties view excerpt (the lower part of Figure 7.8) shows the profile specifying the roles and rights.

A birth certificate is issued by the *issueBirthCertificate* operation of the *MACS* class. To model an access control on this operation, the «`abacRequire`» stereotype is used, showing an excerpt in Figure 7.9. To execute this operation, the *modify* right is required. With respect to the properties view in Figure 7.8, this right is only as-

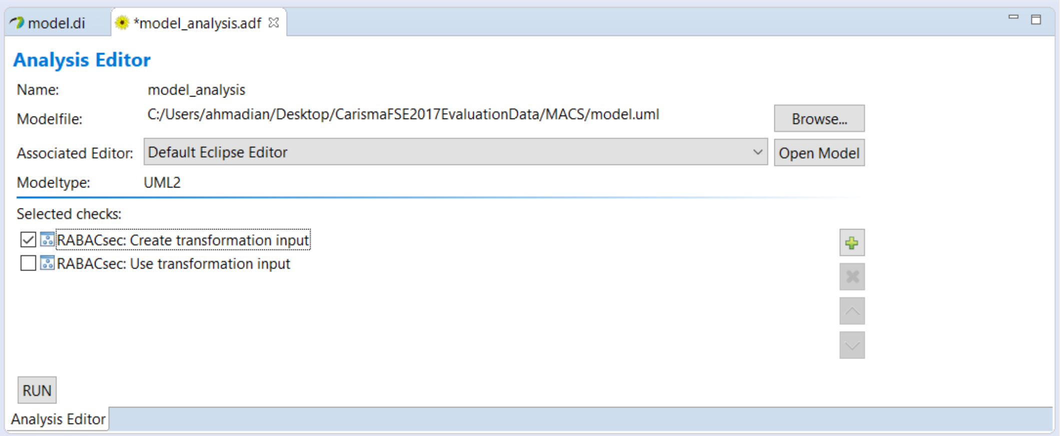


issueBirthCertificate () : BirthCertificate

Applied stereotypes:

abacRequire (from RABAC)
right: String [1] = modify
filters: String [0..1] = Status='Submitted'

Figure 7.9: A properties view excerpt showing the profile defined for the *issueBirthCertificate* operation of the *MACS* class



Analysis Editor

Name: model_analysis

Modelfile: C:/Users/ahmadian/Desktop/CarismaFSE2017EvaluationData/MACS/model.uml

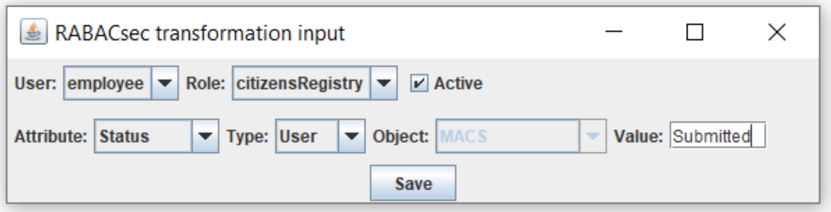
Associated Editor: Default Eclipse Editor

Modeltype: UML2

Selected checks:

- ☑ RABACsec: Create transformation input
- □ RABACsec: Use transformation input

Figure 7.10: A screenshot showing the RABAC check. First, a configuration file has to be generated (*Create transformation input*). Afterwards, an analysis is performed (*Use transformation input*).



User: employee Role: citizensRegistry ☑ Active

Attribute: Status Type: User Object: MACS Value: Submitted

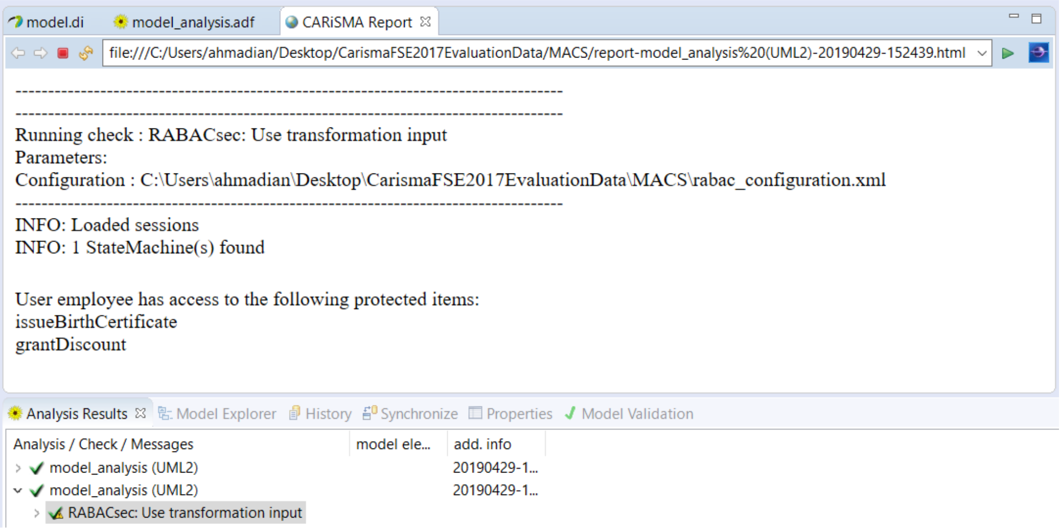
Figure 7.11: The RABAC pop up menu to insert required input (including a user and an attribute) for a RABAC analysis

signed to an employee. Furthermore, the *status* attribute has to be set to *Submitted*.

To perform a RABAC analysis, first, a configuration file has to be generated (Figure 7.10, *Create transformation input*). The configuration file specifies the user and the attribute, that is analyzed. For instance, as shown in Figure 7.11, we set the user to *employee*, the role to *citizenRegistry*, and the *status* attribute to *Submitted*.

Eventually, we can perform a RABAC analysis (Figure 7.10, *Use transformation input*). An excerpt of the analysis result is demonstrated in Figure 7.12. Concerning the annotations in the class diagram, an employee has access to two operations, namely *issueBirthCertificate*, and *grantDiscount*. In a privacy analysis such information is used to further verify whether an employee is authorized—is an element of to the privacy preferences (visibility lattice)—to issue a birth certificate that contains the AMKA (SSN) of a citizen.

The results of an analysis include detected security and privacy design violations. Different actions may be performed on such analysis results. In the context of the


model.di
model_analysis.adf
CARiSMA Report
file:///C:/Users/ahmadian/Desktop/CarismaFSE2017EvaluationData/MACS/report-model_analysis%20(UML2)-20190429-152439.html
Analysis Results
Model Explorer
History
Synchronize
Properties
Model Validation

```
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
Running check : RABACsec: Use transformation input
Parameters:
Configuration : C:\Users\ahmadian\Desktop\CarismaFSE2017EvaluationData\MACS\rabac_configuration.xml
--------------------------------------------------------------------------------
INFO: Loaded sessions
INFO: 1 StateMachine(s) found

User employee has access to the following protected items:
issueBirthCertificate
grantDiscount
```

Analysis / Check / Messages	model ele...	add. info
✓ model_analysis (UML2)		20190429-1...
✓ model_analysis (UML2)		20190429-1...
✓ RABACsec: Use transformation input		

Figure 7.12: An excerpt of the analysis result following performing a RABAC analysis

VPP, the analysis results are contained in the agreements, that are concluded between citizens and PAs on the use of the personal data of the citizens.

7.2.3 Privacy Level Agreements within the VisiOn Privacy Platform

In Section 3.4, to capture the privacy preferences of a data controller, we proposed to use PLAs. PLAs facilitate the application of modular privacy analysis introduced in Section 4.3.1. Following Section 5.3.6, PLAs are further used to document a privacy impact assessment report, thereby tracking privacy threats, risks and privacy controls to mitigate the arising risks. Moreover, according to Chapter 6, after choosing privacy features (including privacy design strategies, privacy design patterns and PETs) to enhance a system model, they have to be incorporated in a PLA. We updated the PLA outline (Appendix C) to cover the differences (Appendix B) between the GDPR and the former data protection regulation of the EU (on which the PLA outline introduced by the Cloud Security Alliance relies). In Figure 3.9, we presented a metamodel to specify the structure of a PLA.

As previously mentioned, one of the functionalities of the VPP is to produce agreements on the use of personal data. The VPP [69] establishes agreements between citizens and public administrations to:

- Handle personal data and keep tracking of data controllers privacy needs.

- Describe the personal data processing.
- Keep track of identified threats and suggested controls to mitigate arising risks.

Figure 7.13 presents a PLA instance [68] produced by the VPP between a citizen and the Municipality of Athens (MoA). We contributed to develop the structure of such PLAs during the VisiOn project. Several sections of the PLA produced by the VPP correspond to the elements of our proposed PLA structure.

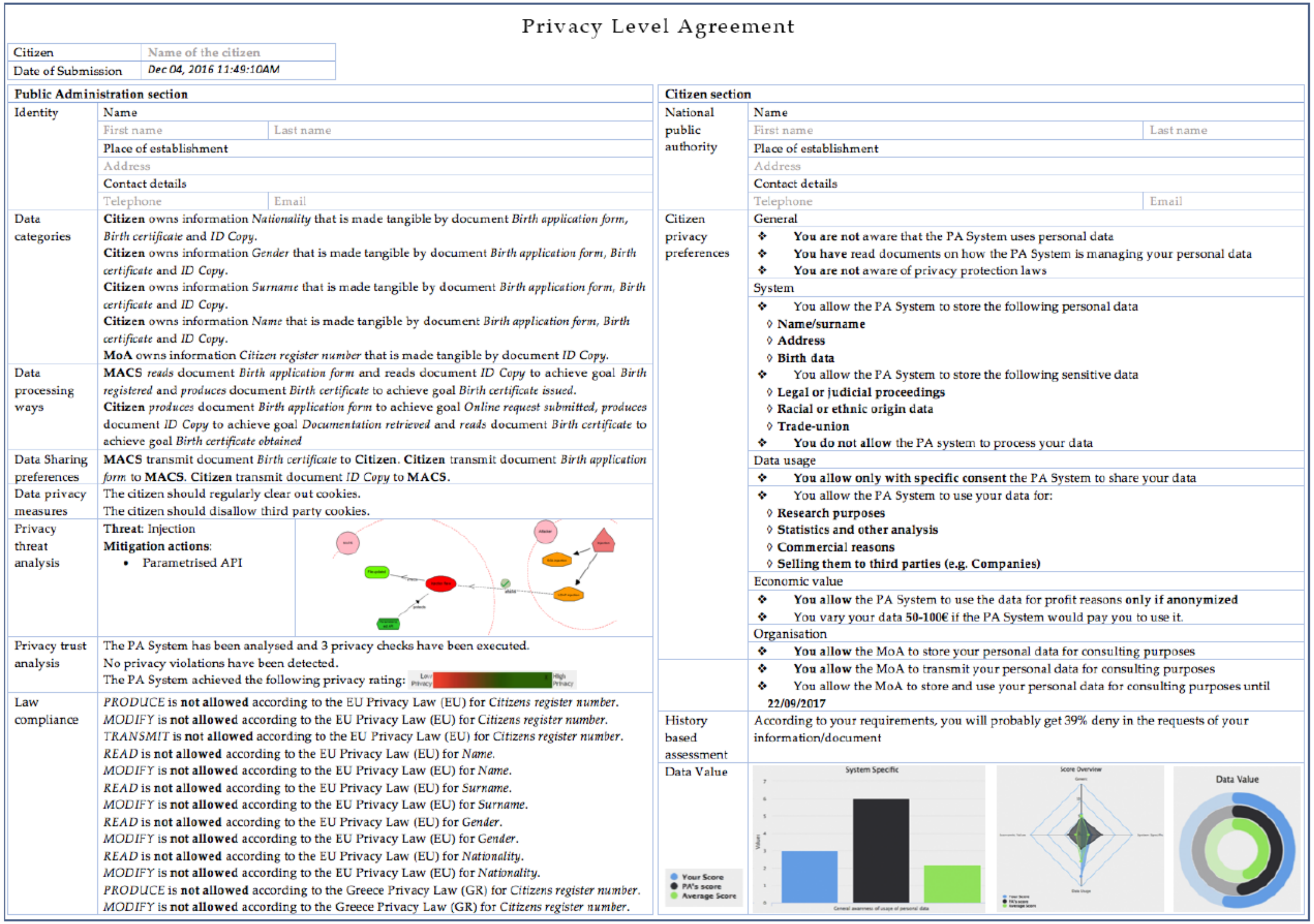
The PLAs that are generated by the VPP rely on the results of several tools. Therefore they include several sections that are not covered in our metamodel demonstrated in Figure 3.9. Particularly, the *History based assessment* and *data value* sections are not relevant for our model-based privacy by design methodology.

All the tools that are integrated into the VPP (including CARiSMA) stores the results of their analysis in the VDB. A PLA is generated by compiling such results stored in the VDB. According to Figure 7.13, in the *privacy trust analysis* section, the results of performing a CARiSMA analysis are presented by specifying potential privacy design violations. In the VPP, a tool (SecTro [149, 166]) is used to analyze security requirements and identify the potential privacy threats. In a PLA, the results of this tool are captured in the *privacy threat analysis* section.

The *Data categories* and *Data processing ways* sections correspond to two classes *personal data* and *process* of our metamodel, respectively. However, we defined a thorough structure for a process by identifying the associated operations that process personal data and their objectives to process personal data.

The preferences to process personal data are specified in two sections in Figure 7.13, namely *Law compliance* and *Citizen privacy preferences*. The assertions that are listed in *Law compliance* are derived from regulations. *Citizen privacy preferences* specify the collected personal data of a citizen and the authorized purposes to process personal data. These assertions and preferences can be analyzed by our model-based privacy analysis methodology (given an annotated system model). Our definition of privacy preferences in the PLA metamodel (Figure 3.9) covers these two sections. In our PLA structure, we further require an authorized granularity level and a retention condition to process a piece of personal data.

According to Figure 7.13, the *Data privacy measure* section indicates the appropriate measures (for instance, PETs) to ensure the privacy of personal data in a PA. This section corresponds to the *control* element in our proposed PLA structure. Concerning the metamodel introduced in Figure 6.6, which demonstrates the underlying concepts of the privacy enhancement, we defined a more rigorous structure for the



Privacy Level Agreement

Citizen	Name of the citizen
Date of Submission	Dec 04, 2016 11:49:10AM

Public Administration section		
Identity	Name	
	First name	Last name
	Place of establishment	
	Address	
	Contact details	
	Telephone	Email
Data categories	**Citizen** owns information *Nationality* that is made tangible by document *Birth application form*, *Birth certificate* and *ID Copy*. **Citizen** owns information *Gender* that is made tangible by document *Birth application form*, *Birth certificate* and *ID Copy*. **Citizen** owns information *Surname* that is made tangible by document *Birth application form*, *Birth certificate* and *ID Copy*. **Citizen** owns information *Name* that is made tangible by document *Birth application form*, *Birth certificate* and *ID Copy*. **MoA** owns information *Citizen register number* that is made tangible by document *ID Copy*.	
Data processing ways	**MACS** *reads* document *Birth application form* and reads document *ID Copy* to achieve goal *Birth registered* and *produces* document *Birth certificate* to achieve goal *Birth certificate issued*. **Citizen** *produces* document *Birth application form* to achieve goal *Online request submitted*, *produces* document *ID Copy* to achieve goal *Documentation retrieved* and *reads* document *Birth certificate* to achieve goal *Birth certificate obtained*	
Data Sharing preferences	**MACS** transmit document *Birth certificate* to **Citizen**. **Citizen** transmit document *Birth application form* to **MACS**. **Citizen** transmit document *ID Copy* to **MACS**.	
Data privacy measures	The citizen should regularly clear out cookies. The citizen should disallow third party cookies.	
Privacy threat analysis	**Threat**: Injection **Mitigation actions**: • Parametrised API	
Privacy trust analysis	The PA System has been analysed and 3 privacy checks have been executed. No privacy violations have been detected. The PA System achieved the following privacy rating: Low Privacy – High Privacy	
Law compliance	*PRODUCE* is **not allowed** according to the EU Privacy Law (EU) for *Citizens register number*. *MODIFY* is **not allowed** according to the EU Privacy Law (EU) for *Citizens register number*. *TRANSMIT* is **not allowed** according to the EU Privacy Law (EU) for *Citizens register number*. *READ* is **not allowed** according to the EU Privacy Law (EU) for *Name*. *MODIFY* is **not allowed** according to the EU Privacy Law (EU) for *Name*. *READ* is **not allowed** according to the EU Privacy Law (EU) for *Surname*. *MODIFY* is **not allowed** according to the EU Privacy Law (EU) for *Surname*. *READ* is **not allowed** according to the EU Privacy Law (EU) for *Gender*. *MODIFY* is **not allowed** according to the EU Privacy Law (EU) for *Gender*. *READ* is **not allowed** according to the EU Privacy Law (EU) for *Nationality*. *MODIFY* is **not allowed** according to the EU Privacy Law (EU) for *Nationality*. *PRODUCE* is **not allowed** according to the Greece Privacy Law (GR) for *Citizens register number*. *MODIFY* is **not allowed** according to the Greece Privacy Law (GR) for *Citizens register number*.	

Citizen section		
National public authority	Name	
	First name	Last name
	Place of establishment	
	Address	
	Contact details	
	Telephone	Email
Citizen privacy preferences	General ❖ **You are not** aware that the PA System uses personal data ❖ **You have** read documents on how the PA System is managing your personal data ❖ **You are not** aware of privacy protection laws	
	System ❖ You allow the PA System to store the following personal data ◊ **Name/surname** ◊ **Address** ◊ **Birth data** ❖ You allow the PA System to store the following sensitive data ◊ **Legal or judicial proceedings** ◊ **Racial or ethnic origin data** ◊ **Trade-union** ❖ **You do not allow** the PA system to process your data	
	Data usage ❖ **You allow only with specific consent** the PA System to share your data ❖ You allow the PA System to use your data for: ◊ **Research purposes** ◊ **Statistics and other analysis** ◊ **Commercial reasons** ◊ **Selling them to third parties (e.g. Companies)**	
	Economic value ❖ **You allow** the PA System to use the data for profit reasons **only if anonymized** ❖ You vary your data **50-100€** if the PA System would pay you to use it.	
	Organisation ❖ **You allow** the MoA to store your personal data for consulting purposes ❖ **You allow** the MoA to transmit your personal data for consulting purposes ❖ You allow the MoA to store and use your personal data for consulting purposes until **22/09/2017**	
History based assessment	According to your requirements, you will probably get 39% deny in the requests of your information/document	
Data Value	System Specific Score Overview Data Value ● Your Score ● PA's score ● Average Score	

Figure 7.13: A PLA instance [68] generated by the VPP in the context of the VisiOn project with respect to the MoA case study. Different tools that are integrated into the VPP provide various information to generate a PLA.

controls including privacy design strategies, privacy patterns and PETs.

The *Vito* tool (see Figure 7.2) provides an interface to show the established PLAs to the VPP users (PAs and citizens). In Section 4.4, we mentioned that our industry partners (PAs) evaluated the VPP and documented the results in *Deliverable 5.2* [43]. Concerning the usefulness of the PLAs, OPBG (a PA) reported that:

> "More than 80% of the healthcare users stated that the PLA of the VisiOn privacy platform offers a complete insight on privacy and security issues".

The OPBG's survey to evaluate the functionalities of the VPP, particularly, involved one question on the usefulness of the PLAs: "Do you think the section where you can view your PLAs offers a complete insight into the VisiOn approach on privacy and security issues?" The respondents (in total 99 users) were citizens (89 users) and hospital administrators (10 users). Since the other two PAs did not directly evaluate the usefulness of the PLAs in their questionnaires, we did not show any relevant results.

As mentioned earlier in this section, using the VPP, the PLAs are established between citizens and PAs. In contrast, in our model-based privacy by design methodology, PLAs are established between data controllers and data processors, where both are organizations. Our definition of the PLAs follows the scope of the PLA outline introduced by the CSA, where only business-to-business scenarios are considered. We further showed that our proposed PLA structure covers the PLA structure developed within the VisiOn project. We defined more rigorous structures to specify processes, personal data categories, privacy preferences, and privacy controls.

7.2.4 Implementation and Availability

The CARiSMA tool suite is based on the Eclipse IDE and consists of several components. In Figure 7.14, a UML component diagram illustrates the main components of CARiSMA. This figure further demonstrates the components that are added to the architecture of CARiSMA to support the newly added functionalities. The two crucial components that enable a security analysis are the *Profiles* and *Checks* components. These two components are extended with the privacy as well as *rabac* profiles and the privacy checks (currently rabac).

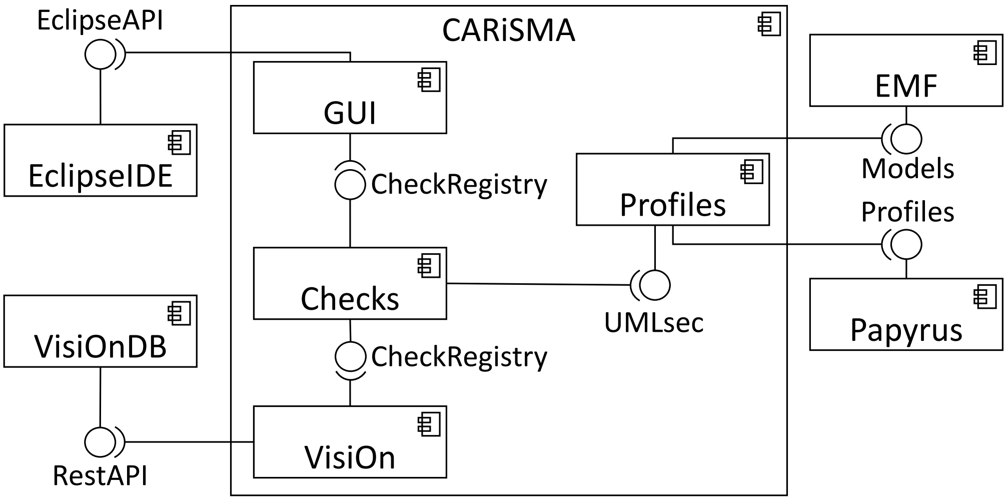


Figure 7.14: The architecture of CARiSMA extended by the relevant VisiOn components

The *Profiles* component contains the specifications of the UML profiles, and registers them to two external components, namely *Papyrus* and *EMF* model registry. Papyrus[6] is a UML tool used to model the systems. However, CARiSMA is able to work on any EMF-based UML model. EMF (Eclipse Modeling Framework) [191] is a modeling framework for building tools and other applications based on structured data models[7]. Therefore, the profiles component is additionally registered to the EMF component. This allows usage of the security profile in model transformation tools or with the EMF-based *OCL* (*object-oriented language*) [156] implementations.

The *privacy*, *rabac* and *UMLsec* profiles enable different security and privacy checks, which are implemented in a *Checks* component. This component provides the interface *CheckRegistry* to allow other components to access available checks, to execute those on UML models, and to generate analysis results. These checks use the interface *UMLsec* provided by the *Profiles* component to verify whether the security and privacy requirements are supported by a system model.

Using the *CheckRegistery* interface, the *GUI* component provides a user interface for executing checks and displaying the results. This component leverages the *EclipseAPI* interface for integrating the CARiSMA user interface with Eclipse.

The *VisiOn* component provides interfaces to the other tools such as STS. The inte-

[6] `https://www.eclipse.org/papyrus/` (accessed: 2019-06-01)

[7] `https://www.eclipse.org/modeling/emf/` (accessed: 2019-06-01)

gration with STS is enabled by the VisiOn database. The component is connected to *VisiOn database* component, which is accessed over a *RestAPI*. STS models are stored in this database and CARiSMA retrieve these models from the database and perform a pre-analysis.

CARiSMA, including the Vision Extension, is published under the Eclipse Public License (EPL) and may be installed from the update-site[8]. Additional help content such as installation instructions and screencasts are available on the CARiSMA website. In a screencast[9], we additionally demonstrate and describe the newly added functionalities that were introduced in this section. In Appendix G, we provide more information on the screencast.

7.2.5 Related Work

There are several approaches to support model-based security analysis. Some of those are summarized and discussed by Lano et al. [139]. The model-based use of security patterns has been addressed by some research [152]. Further research makes use of aspect-oriented modeling for model-based security [89]. Heitmeyer et al. propose the application of formal methods on minimal state machine models for security verification [104].

SecureUML provides a role-based access control using UML models [141]. While CARiSMA provides interfaces for adding arbitrary profiles and checks, SecureUML is limited to access control.

The CORAS tool provides security risk analysis [64]. CORAS works on proprietary models and uses the CORAS language, which was originally a UML profile but later defined as a domain specific language.

In the VisiOn project, two tools, namely, SecTro and JTrust, are integrated within the requirement analysis component to provide security threat analysis. SecTro is built upon the Secure Tropos approach and is used to model security during requirements engineering [149, 166]. JTrust evaluates the trustworthiness of a system based on trust and control models [167].

Islam et al. integrated the Secure Tropos approach with UMLsec [117], to support the alignment of secure software engineering with legal regulations. However, this work does not support privacy requirements and they do not analyze security requirements to automatically perform appropriate UMLsec checks.

[8] `http://carisma.umlsec.de/updatesite` (accessed: 2019-06-01)

[9] `https://youtu.be/b5zeHig3ARw` (accessed: 2019-06-01)

Furthermore, in [14, 68, 70], the authors—our partners in the VisiOn project—describe the integration of their tools and their contributions to the VPP.

7.2.6 Preliminary Conclusion

We have introduced several new functionalities to support privacy in CARiSMA. Through analyzing a set of given privacy and security requirements, CARiSMA assists a system developer to express security and privacy requirements within models. Specifically, through a pre-analysis of such requirements, a help report is generated to assist a system developer to annotate a system model with corresponding UML profiles, and a CARiSMA analysis is automatically initialized to perform several privacy and security checks. We further introduced RABAC which provides a prototypical tool support to perform a privacy analysis.

We mainly focused on the interaction between CARiSMA and the STS-Tool within the VPP. However, CARiSMA potentially might interact with other available tools in the VPP. The *Data Value Tool* (*DVT*), integrated into the *web framework* of the VPP (Figure 7.2), assesses the value of the citizens' personal data. Using the information obtained from the citizens and the PAs (through questionnaires), the DVT compares several perceptions on the personal data values (such as data footprint, economic value, and data conflicts) to identify the risks and the importance of processing of personal data [71]. Our *privacy impact assessment* methodology (Chapter 5) may benefit from the DVT to identify the risks. In an impact assessment (Section 5.3.4), two *Impact Value*s relying on the protection demands of two stakeholders (a data controller, and a data processor) contribute to assess the emerging risks in a system model. The values assessed by the DVT may be used to enhance the estimation of the *Impact Value*s.

7.3 Pattern-Based Risk Analysis with the ClouDAT Framework

In the previous section, we introduced CARISMA that allows one to perform privacy and security analysis on annotated system models. In Chapter 5, we proposed a privacy impact assessment (*PIA*) methodology to identify privacy risks relying on the results of model-based privacy and security analysis. In this section, we introduce the ClouDAT framework that is originally developed to perform a risk analysis in cloud environments. We explain how ClouDAT framework can be used to conduct a *PIA*, identify privacy risks and suggest appropriate controls. This section benefits from the material provided in [5, 12, 195].

The utilization of cloud computing services has been ever growing in the past years and the growth of such services is expected to continue in the near future [16]. The National Institute of Standards and Technology describes cloud computing as "ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction" [145]. However, the acceptance of cloud computing is growing slowly, due to the fact that cloud computing introduces new threats.

A possible way to encounter skepticism and raise acceptance is the certification of cloud providers according to standards such as ISO 27001 [119]. However, for small and medium-sized enterprises (SMEs), offering cloud solutions is a rather complex task, due to the lack of know-how and resources to conduct an ISO27001 compliant risk assessment and generate the appropriate documentation to reach the certification. The ClouDAT project[10] offers a framework for helping SMEs handling the certification process. The framework contains a risk assessment process and allows the automatic generation of ISO27001 compliant documentation based on the outcomings of the risk assessments [12].

The ClouDAT framework is an open source framework and supports SMEs to conduct the certification of the cloud services. Generally, the ClouDAT framework establishes an Information Security Management System (ISMS) based on the ISO 27001 [119] standard. The development of an ISMS allows organizations to implement a framework to manage the security of their information assets such as financial or customer information.

The framework benefits from several artifacts:

- A metamodel for the risk analysis process complying with ISO 27001 standard,
- A catalog of security requirements,
- A catalog of cloud-specific threats,
- A catalog of security controls.

We contributed to devising the risk treatment method of the ClouDAT framework and establishing the catalog of security controls.

[10] `http://ti.uni-due.de/ti/clouddat/de/` (accessed: 2019-06-01)

7.3.1 The Overview of the ClouDAT Risk Analysis Process

Figure 7.15 [12] presents an overview of the ClouDAT risk analysis process, which complies with ISO 27001 standard. This figure is simplified in a way that the artifact flow from the output of a step to the following step is not demonstrated. The process includes the following steps [12]:

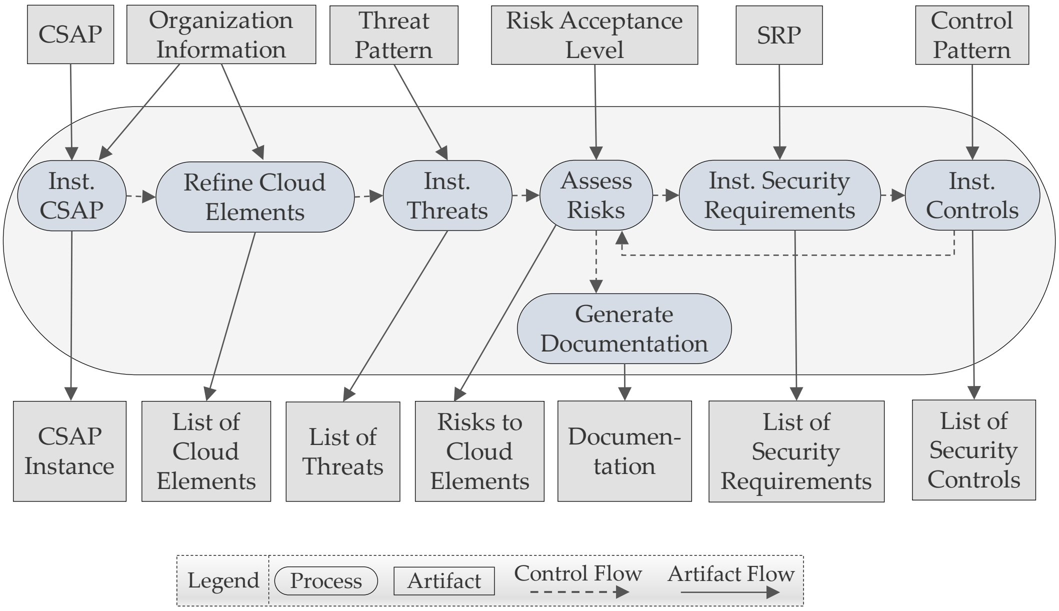


Figure 7.15: The overview of the pattern-based risk analysis process introduced in [12] (Inst. stands for the instantiate).

Instantiate CSAP. In this step, the scope and the boundaries of the ISMS are defined. To this end, the *Cloud System Analysis Pattern* (*CSAP*) [25] is employed. The CSAP provides a structured approach to describe cloud environments. It enables one to model cloud elements (the elements of a cloud environment) and the relations between such elements. Cloud elements are the central element in the ClouDAT framework. A cloud element may basically be anything of value to the company, such as documentation or a real physical system. A cloud element is identified by a unique name und contains additional information such as type, owner, descriptions, and a location. The output of this step is an instance of a CSAP which indicates a set of cloud elements.

Refine Cloud Elements. In this step, the cloud elements that are important to the risk analysis have to be determined. Such elements provide a basis to perform the risk analysis. The results of this phase are collected in a table, which is called *cloud element list*. This table contains all mandatory cloud elements for the risk analysis. The cloud elements refinement is performed in two steps [12]:

- Refine cloud elements and their location: The abstract mandatory cloud elements are refined into more concrete and detailed cloud elements. This step requires the organization (under assessment) information such as work instructions and organigrams.
- Assign responsibilities and relationships: The responsibilities of the cloud elements are identified, and the relations between the cloud elements are determined.

Instantiate Threats. In this step, a threat analysis of the elements contained in the cloud element list is performed. A threat analysis verifies whether a cloud element has violations that may be exploited by a threat. The ClouDAT framework provides a catalog of predefined threats and violations for the cloud elements. This catalog is based on earlier works [46, 77, 103] and the list of cloud computing top threats [47] informed by the *CSA*.

Assess Risks. In this step, the existing risks to the cloud elements are assessed. Prior to a risk analysis, a risk assessment approach and a risk acceptance level must be specified. Generally, the risk assessment relies on the business impact and the security failures. Business impacts express the consequences that affect the failure of the security goals. Furthermore, considering the identified threats and the violations, the likelihoods of potential security failures for all cloud elements have to be specified. The risk levels of the cloud elements are determined by multiplying the likelihoods of the security failures and the assigned values to the business impacts. The risk assessment corresponds to Section 4.2.1 of the ISO 27001. By comparing the risk levels of the cloud elements with the predefined risk acceptance level, the cloud elements that are in danger (have risk levels higher than the risk acceptance level) and require appropriate risk treatments, are identified.

Instantiate Security Requirements. In this step, a risk treatment method to reduce the risks of the cloud elements that are in danger is identified. If a cloud element has an unacceptable risk level, the corresponding security requirements that are at risks, have to be identified. To this end, *security requirement pattern*s (*SRP*s) are used [27, 28]. In a concrete certification process, security requirement patterns are instantiated, and for each cloud element with an unaccepted risk level, a security requirement will be defined. ClouDAT framework provides a catalog of predefined SRPs.

Instantiate Controls. To mitigate the risks, appropriate security controls have to be applied. The security controls are represented by *control patterns* (*CP*). Additionally, a catalog of predefined security controls is provided. This catalog is introduced in Section 7.3.3.1. According to Figure 7.15, after identifying and applying the appropriate security controls, the risk assessment process may be iteratively continued to verify whether the risks are appropriately mitigated.

Generate Documentation. In the final step, a document is generated. This document contains the list of refined cloud elements, the list of threats and corresponding violations, the list of cloud elements with unaccepted risk level, the list of security requirements, and finally the list of selected controls to mitigate the identified risks. The resulting documentation is used as a foundation for the certification.

7.3.2 The CLouDAT's Risk Analysis Metamodel

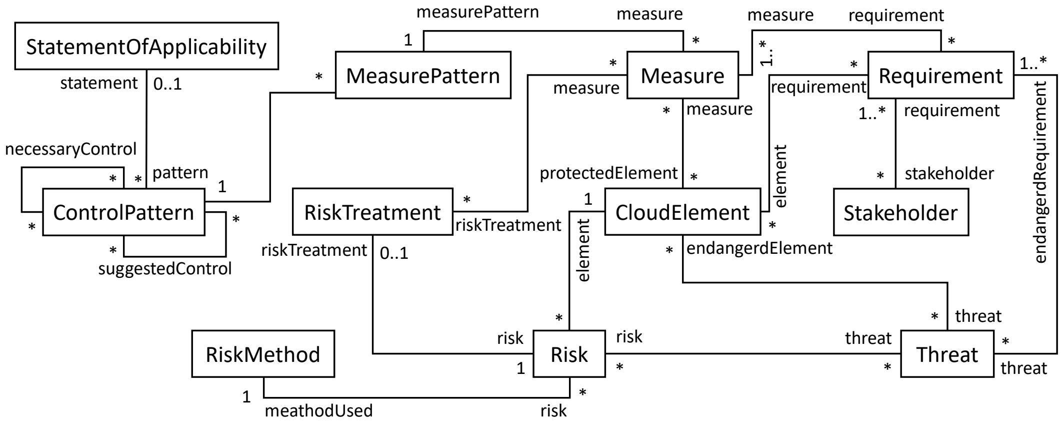


Figure 7.16: Excerpt of the risk analysis metamodel [195]

In order to describe the underlying concepts of the ClouDAT risk analysis process, an excerpt of the risk analysis metamodel is demonstrated in Figure 7.16. This metamodel only presents the key concepts. In a risk analysis, the main aim is to identify the risks that affect the *CloudElement*s, elicited in the *cloud elements identification* step.

CloudElements are subject to the *requirement*s specified by *stakeholder*s. The requirements are expressed using the ClouDAT's predefined *RequirementPattern*s illustrated in Figure 7.17. They are composed of *generic* and *fixed text passage*s. Fixed text passages represent the meaning of a security requirement and cannot be edited by the user. Generic text passages may include *multi-selection*s or relations to specific cloud elements. In order to instantiate a certain requirement, the blank texts in a requirement pattern have to be filled out [25, 27, 28].

Figure 7.18 presents a requirement pattern. It consists of fixed text and multi-selections. The elements in squared brackets represent the different options for

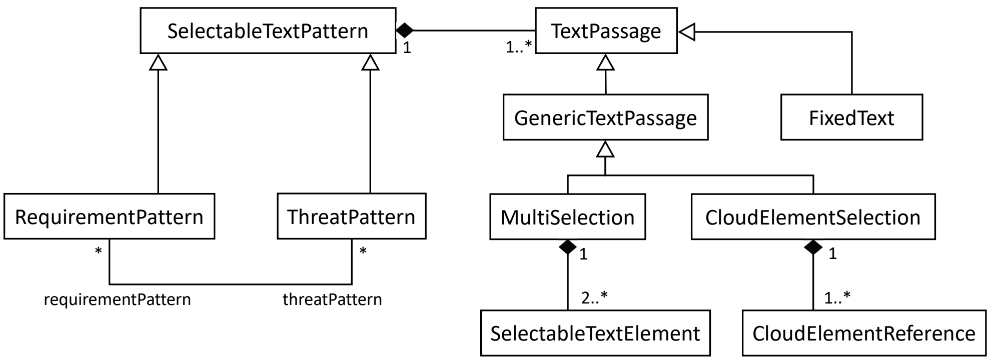
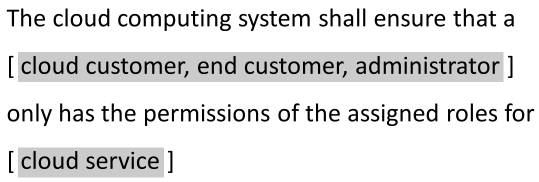


Figure 7.17: Excerpt of the selectable text metamodel [195]. The metamodel demonstrates the element of a requirement or a threat pattern.

a multi-selection. The rest (not in squared brackets) is fixed text. According to Figure 7.16 requirements can be endangered by *threat*s. Similar to the requirements, a threat may be instantiated from a *ThreatPattern*. The structure of ThreatPatterns is demonstrated in Figure 7.17 as well. Figure 7.19 shows a threat pattern.

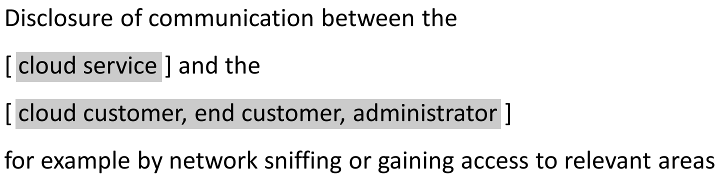

The cloud computing system shall ensure that a

[cloud customer, end customer, administrator]

only has the permissions of the assigned roles for

[cloud service]

Figure 7.18: An example of a security requirement pattern


Disclosure of communication between the

[cloud service] and the

[cloud customer, end customer, administrator]

for example by network sniffing or gaining access to relevant areas

Figure 7.19: An example of a threat pattern

A risk represents the potential that a given threat will exploit violations of a cloud element or group of cloud elements and thereby causing harms to the organization [121]. A risk consists of likelihoods, business impacts and the resulting risk levels for the protection goals: confidentiality, integrity, availability. Since a certification requires every risk to be handled or accepted, it is mandatory to deliver an acceptance rule for all the identified risks. The acceptance rule is identified by *RiskMethod* (Figure 7.16). The risks exceeding the acceptance level have to be treated. For such a treatment, ClouDAT defines *RiskTreatments*. A risk treatment comprises several *Measure*s. In the following section, we elaborate on the rist treatment method of the ClouDAT framework.

7.3.3 The ClouDAT's Risk Treatment Method

The ClouDAT's risk treatment method complies with the ISO 27001 and includes four treatment methods: **(I)** applying appropriate controls, **(II)** accepting risks, **(III)** avoiding risks, and **(IV)** transferring the associated business risks to other parties. To ensure that a risk is mitigated by applying controls, the measures that were used to reduce the risk has to be specified. ClouDAT distinguishes between controls and measures. A control describes an action that has to be taken to reduce a risk, however, a control is generally too abstract. A measure provides the detailed specification of a control. For instance, for the *asymmetric encryption* control, a specific measure such as the *RSA* (*Rivest–Shamir–Adleman*) algorithm may be suggested. As demonstrated in Figure 7.16, the controls are specified by the *ControlPattern* class. Controls may suggest the use of other controls or require the implementation of other controls. A *MeasurePattern* provides the implementation possibilities of a control. An instance of a MeasurePattern is a *Measure*, that is assigned to a set of requirements and cloud elements.

The risk treatment method benefits from a catalog (list) of security controls. Initially, to mitigate the risks, the necessary controls have to be determined. Afterwards, a comparison of the determined controls with those in the ISO 27001 must be performed, verifying that no mandatory controls have been excluded. Eventually, a statement of applicability that incorporates the mandatory controls and explanations for inclusions and exclusions of the controls must be provided. In the following section, we introduce the ClouDAT's control list.

7.3.3.1 The Structure of the Control List

To mitigate the risks by applying appropriate controls, we provide a control list, showing an excerpt in Table 7.2. Each control has a set of aspects, which are specified below. Table 7.2 does not show all the aspects of each control.

- **ID:** The controls documented in the control list are generally derived from the security controls provided in ISO 27001 [119]. Similar to the ISO document, A unique identification number (*ID*) is used to identify each control. Furthermore, to support all the security requirements that are used in ClouDAT, we included a set of additional controls identified in earlier work. A set

Table 7.2: An excerpt of the ClouDAT's control list

ID	Control	Dependencies	Requirement	BSI Reference
A.5.1.2	Review of the information security policy	Necessary: 5.1.1	Security management	M2.192, B1.0, M2.335, M2.1
A.9.4.3	Password management system	Necessary: 5.1.1 Suggested: A.9.3.1	Confidentiality, Security management	M 2.11, M 4.133
A.14.3.1	Protection of test data	–	Confidentiality 1, Confidentiality 2	M 2.83
A.18.1.4	Privacy and protection of personally identifiable information	Necessary: A.18.1.1	Authenticity 1, Security management 6, Privacy	B 1.16, B 1.5, M 3.2, M 2.10, M 2.205
A.18.1.5	Regulation of cryptographic controls	Necessary: A.18.1.3	Security management (6, 8, 9, 10), Integrity, Confidentiality	B 1.16, M 2.163
SP.35.8	Secure model-view-controller	Suggested: A.12.1.4, SP.28.2	Authenticity, Integrity	[80]

of security patterns [80, 183] are also contained in the control list. The ID of such patterns are started with the *SP*. During the development of the VPP, we added the NIST privacy controls [154] to our control list.

- **Control:** All the controls are identified by a short title. Concerning ISO controls, the title matches the one in the original document. The rest of the controls are labeled in a similar manner.

- **Dependencies:** Two types of dependencies between the controls are identified.

  - Necessary: The depending control has to be implemented as well in the most cases. If the user chooses not to apply the necessary control, the reason must be justified.
  - Suggested: The depending control might be useful to support the current control or its measure. The tool offers these controls as an option to the user.

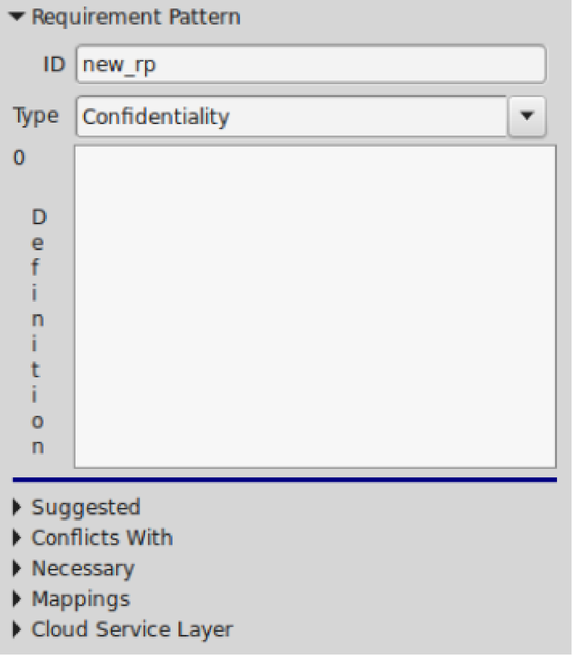
- **Requirement:** The complete list of the requirements are provided in [195]. Such requirements serve as a basis for the security requirement patterns (Section 7.3.2, Figure 7.17), and relies on the basic list of security requirements that are introduced in [101].
- **Instance Type:** The instance type of the control (if available).
- **Additional Consideration:** The necessary cloud elements to perform control with relevant security aspect. The implementation of a control can lead to the creation of additional cloud elements, that have to be protected accordingly.
- **BSI References:** The related entries from the BSI Grundschutz catalogs (IT Baseline Protection Catalogs) [39].
- **CCM References:** List of similar controls from *CCM* (*Cloud Control Matrix*) [51]. CCM is a control list provided by *CSA*.
- **Technology/Organization:** Each control is classified whether it is primarily (+) or supportively (∼) technical or organizational.
- **Description of control:** A textual description of the control.

7.3.3.2 Risk Treatment Process

According to Section 7.3.1, for the cloud elements with unaccepted risk level, appropriate security requirements are elicited. Concerning the control list (Table 7.2), each control is mapped to a set of security requirements, thereby specifying how the controls fulfill the associated security requirements. Thus, the elicited security requirements determine the necessary controls to reduce the risks. The process of determining the necessary controls has to consider the dependencies between the controls.

After the selection of the controls, we need to verify whether the risk levels of the cloud elements are reduced. To this end, we need to perform the risk assessment for particular cloud elements to check whether the controls reduce the risk levels or a modification of the controls or other controls are required. This process is iterated until there exist no cloud elements with an unaccepted risk level. However, in certain cases, the risk has to be avoided or ignored. Alternatively, the risk might be transferred to other parties. These decisions are manually made by the security analyzer and must be justified.

Furthermore, a statement of applicability (compliant with Sect. 6.1.3 c-d of the ISO 27001 [119]) has to be provided. To this end, the ClouDAT framework provides



ID new_rp

Type Confidentiality

Figure 7.20: A screenshot, showing a template to create a security requirement pattern using the ClouDAT framework [195]

a template. This template is simply a table, in which for each selected control either a justification has to be provided specifying why the control is excluded, or the overview of the implementation has to be provided, i.e., the necessary and suggested controls to perform the control have to be listed.

For instance, the result of a risk treatment process in an organization may indicate that the *confidentiality* of the personal data might be threatened. Concerning the security requirement patterns (SRP) catalog—introduced in Section 7.3.2—the following pattern exists:

> Confidentiality of personal data of [cloud customer, end customer] shall be achieved.

In Figure 7.20 a screenshot of ClouDAT framework [195] is provided, showing how a SRP is created. A SRP has an ID, a type (in this example, *confidentiality*), a definition including variable and fixed text passages, as well as a set of dependencies and metadata.

To instantiate the security requirement pattern, from the list of identified and refined cloud elements, an element as a representation of the cloud customer or end customer must be inserted into the variable text passage. Consider the scenario, in which *Organization A* is a cloud customer and provides the personal data of its customers. The instantiated requirement is:

> Confidentiality of personal data of *Organization A* shall be achieved.

Using the provided mappings between security requirements and security controls in the control list, we select the relevant control(s):

> To address the security requirement, we apply the controls of the ISO 27001, e.g., access control policy (A.9.1.1), working in secure areas (A.11.1.5), network controls (A.13.1.1), including the controls that are specified as necessary to perform along with mentioned controls.

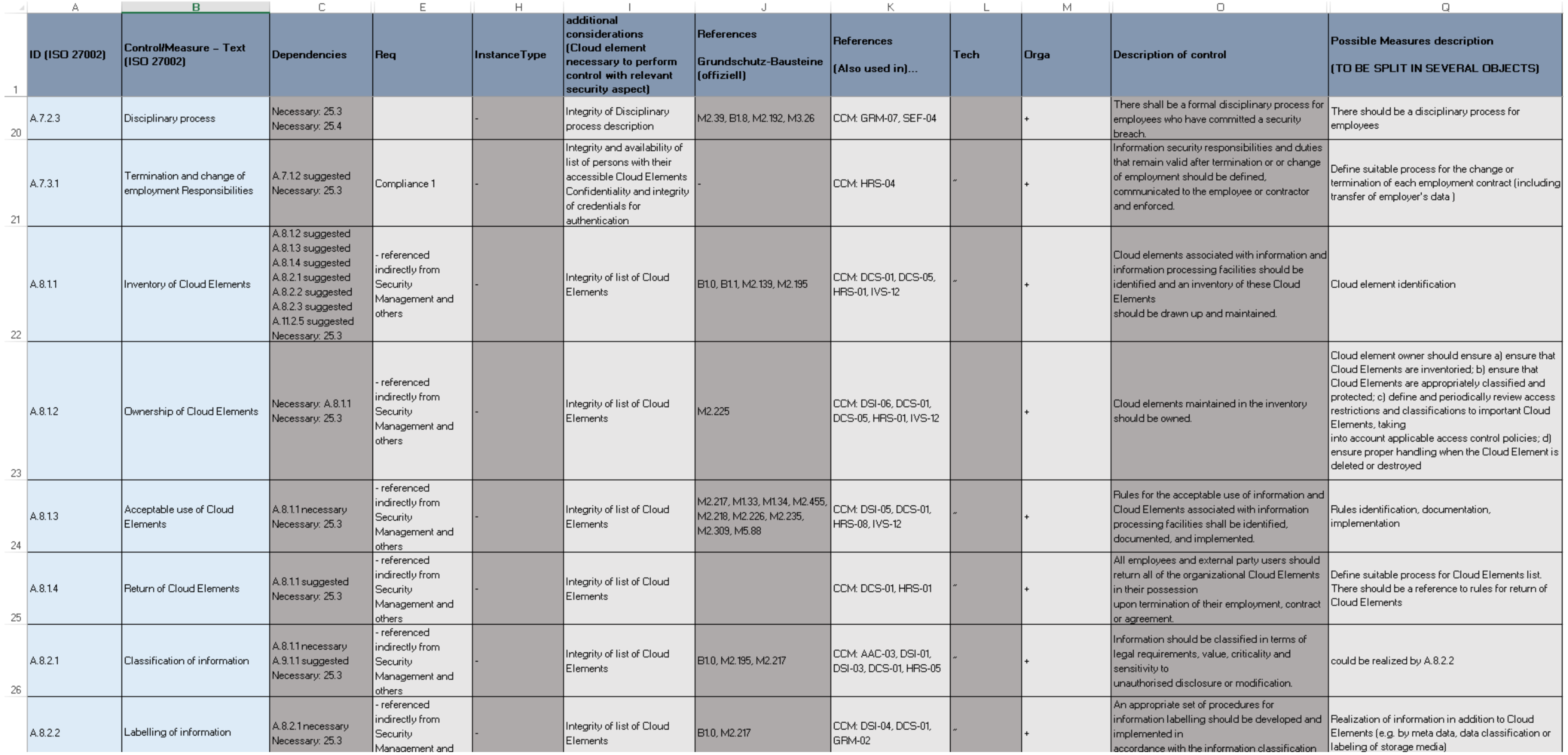
Figure 7.21 shows an excerpt of the excel table containing the controls. In this table, all aspects (Section 7.3.3.1) that are introduced for the controls are included. The full control list can be found online[11].

The final step of the ClouDAT risk analysis process (introduced in Figure 7.15) concerns the ISO 27001 specification, an implementable description of the ISMS. The final documentation uses the results elicited and documented in the previous steps. The results of the step described in this section, namely a list of suggested controls and measures to mitigate the risks, has to be included in the final document of ClouDAT.

7.3.4 Supporting a Privacy Impact Assessment Using the ClouDAT Framework

In Sections 7.3.1-7.3.3, we demonstrated the overall risk assessment process of the ClouDAT framework and described the ClouDAT's risk metamodel and risk treatment process. ClouDAT framework is based on a set of predefined threat and security requirement patterns to assess security risks and identify controls to mitigate the emerging risks. It does not consider a system design to perform a risk assessment. In contrast, our *PIA* methodology relies on performing a model-based privacy and security analysis to identify concrete privacy risks in a system design. However, the correspondence between *PIA* methodology (the six-step presented in

[11] `https://cloud.uni-koblenz-landau.de/s/ocRXY9nJqDWzgpA` (accessed: 2019-06-01)



	A	B	C	E	H	I	J	K	L	M	O	Q
1	ID (ISO 27002)	Control/Measure – Text (ISO 27002)	Dependencies	Req	InstanceType	additional considerations (Cloud element necessary to perform control with relevant security aspect)	References Grundschutz-Bausteine (offiziell)	References (Also used in)...	Tech	Orga	Description of control	Possible Measures description (TO BE SPLIT IN SEVERAL OBJECTS)
20	A.7.2.3	Disciplinary process	Necessary: 25.3 Necessary: 25.4		-	Integrity of Disciplinary process description	M2.39, B1.8, M2.192, M3.26	CCM: GRM-07, SEF-04		+	There shall be a formal disciplinary process for employees who have committed a security breach.	There should be a disciplinary process for employees
21	A.7.3.1	Termination and change of employment Responsibilities	A.7.1.2 suggested Necessary: 25.3	Compliance 1	-	Integrity and availability of list of persons with their accessible Cloud Elements Confidentiality and integrity of credentials for authentication	-	CCM: HRS-04	-	+	Information security responsibilities and duties that remain valid after termination or or change of employment should be defined, communicated to the employee or contractor and enforced.	Define suitable process for the change or termination of each employment contract (including transfer of employer's data)
22	A.8.1.1	Inventory of Cloud Elements	A.8.1.2 suggested A.8.1.3 suggested A.8.1.4 suggested A.8.2.1 suggested A.8.2.2 suggested A.8.2.3 suggested A.11.2.5 suggested Necessary: 25.3	- referenced indirectly from Security Management and others	-	Integrity of list of Cloud Elements	B1.0, B1.1, M2.139, M2.195	CCM: DCS-01, DCS-05, HRS-01, IVS-12	-	+	Cloud elements associated with information and information processing facilities should be identified and an inventory of these Cloud Elements should be drawn up and maintained.	Cloud element identification
23	A.8.1.2	Ownership of Cloud Elements	Necessary: A.8.1.1 Necessary: 25.3	- referenced indirectly from Security Management and others	-	Integrity of list of Cloud Elements	M2.225	CCM: DSI-06, DCS-01, DCS-05, HRS-01, IVS-12		+	Cloud elements maintained in the inventory should be owned.	Cloud element owner should ensure a) ensure that Cloud Elements are inventoried; b) ensure that Cloud Elements are appropriately classified and protected; c) define and periodically review access restrictions and classifications to important Cloud Elements, taking into account applicable access control policies; d) ensure proper handling when the Cloud Element is deleted or destroyed
24	A.8.1.3	Acceptable use of Cloud Elements	A.8.1.1 necessary Necessary: 25.3	- referenced indirectly from Security Management and others	-	Integrity of list of Cloud Elements	M2.217, M1.33, M1.34, M2.455, M2.218, M2.226, M2.235, M2.309, M5.88	CCM: DSI-05, DCS-01, HRS-08, IVS-12	-	+	Rules for the acceptable use of information and Cloud Elements associated with information processing facilities shall be identified, documented, and implemented.	Rules identification, documentation, implementation
25	A.8.1.4	Return of Cloud Elements	A.8.1.1 suggested Necessary: 25.3	- referenced indirectly from Security Management and others	-	Integrity of list of Cloud Elements		CCM: DCS-01, HRS-01	-	+	All employees and external party users should return all of the organizational Cloud Elements in their possession upon termination of their employment, contract or agreement.	Define suitable process for Cloud Elements list. There should be a reference to rules for return of Cloud Elements
26	A.8.2.1	Classification of information	A.8.1.1 necessary A.9.1.1 suggested Necessary: 25.3	- referenced indirectly from Security Management and others	-	Integrity of list of Cloud Elements	B1.0, M2.195, M2.217	CCM: AAC-03, DSI-01, DSI-03, DCS-01, HRS-05	-	+	Information should be classified in terms of legal requirements, value, criticality and sensitivity to unauthorised disclosure or modification.	could be realized by A.8.2.2
	A.8.2.2	Labelling of information	A.8.2.1 necessary Necessary: 25.3	- referenced indirectly from Security Management and	-	Integrity of list of Cloud Elements	B1.0, M2.217	CCM: DSI-04, DCS-01, GRM-02	-	+	An appropriate set of procedures for information labelling should be developed and implemented in accordance with the information classification	Realization of information in addition to Cloud Elements (e.g. by meta data, data classification or labeling of storage media)

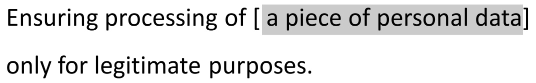
Figure 7.21: An excerpt of the ClouDAT control list

Figure 5.2) and the risk assessment process of ClouDAT (Figure 7.15) gives rise to the fact that one can use ClouDAT framework to support a *PIA*.

The first two steps of the ClouDAT risk assessment process identify a list of cloud elements. In the first step of our *PIA* methodology, we investigate the personal data that are processed in a system model.

Using ClouDAT, the potential threats to the cloud elements are identified using a list of predefined threat patterns. The threats in our *PIA* are identified by evaluating the results of our model-based privacy analysis methodology (Section 5.3.3). After performing a system model analysis, concerning the underlying concept of threat pattern (selectable metamodel), which is introduced in Figure 7.17, a set of privacy threats can be instantiated.

To identify and assess the privacy risks in a *PIA*, we need to identify the privacy targets at risk. The requirement pattern (Figure 7.17) of ClouDAT can be used to specify and instantiate the privacy targets introduced in Table 5.2. In Figure 7.20, we demonstrated a template which is provided by the ClouDAT framework to define requirement patterns. For instance, concerning the privacy target *P1.2 Ensuring processing only for legitimate purposes*, we illustrate a privacy target pattern (requirement pattern) in Figure 7.22. This privacy target pattern comprises a generic text passage (in squared bracket) and a fixed text passage. As stated in Section 5.4.1, the privacy targets may be involved in the questionnaire of the VPP (see Section 7.2.2.1) to estimate the associated protection demands (*impact value*s) of data controllers and data processors for each privacy target.



Ensuring processing of [a piece of personal data]

only for legitimate purposes.

Figure 7.22: A privacy target pattern (similar to a security requirement pattern demonstrated in Figure 7.18).

Moreover, the privacy target *P6.1 Ensuring the confidentiality, integrity, availability, and resilience* is associated with the security requirements. The predefined security requirements catalog of ClouDAT introduces a set of comprehensive security requirements which can be used to improve a *PIA* through performing a more rigorous security assessment.

After identifying the privacy design violations and privacy threats, using Table 5.2, we identify the privacy targets in danger. The ClouDAT risk level estimation is based on the severities and the likelihoods of security threats. In a *PIA*, we ignore the likelihood that a threat may occur. According to Section 5.3.4, we assess the privacy risks (estimate the final *impact assessment* scores) upon the combination of personal data criticality and *impact value*s. The ClouDAT framework uses a predefined risk acceptance level to assess the risk levels. In contrast, a *PIA* uses the *impact assessment* ranges that are introduced in Table 5.5 to categorize the final risk scores.

Finally, in the ClouDAT's risks treatment process, concerning the identified risks to the security requirements and using the ClouDAT's control list, a set of controls and measures are suggested to mitigate the risks. To allow the identification of appropriate controls after performing a *PIA*, the list of security controls is extended by adding the NIST privacy controls. The feature model that we introduced in Section 6.4.1 can be extended by the security controls and measures included in the control list to capture the dependencies between them.

In Section 7.2.3 we illustrated a PLA instance established by the VPP. The final documentation of the ClouDAT framework contains valuable information elicited from the several phases of a risk analysis process such as: violations, threats, risks, and appropriate security controls to mitigate the identified risks. Such information is, in fact, relevant to generate PLAs between organizations. Therefore, several sections of ClouDAT final documentation can be incorporated in PLAs.

Chapter 8

Extending Model-Based Privacy Analysis for the Industrial Data Space by Exploiting Privacy Level Agreements: A Case Study

This chapter shares material with the SAC'18 paper "Extending Model-Based Privacy Analysis for the Industrial Data Space by Exploiting Privacy Level Agreements" [8].

Considering the dramatic impact of the current technology changes on user privacy, it is important to contemplate privacy early on in software development. Ensuring privacy is particularly challenging in industrial ecosystems, where a data processor may depend on or cooperate with other data processors to provide an IT service to a service customer. An example of such ecosystems is the *Industrial Data Space* (*IDS*). The IDS provides a basis for creating and using smart IT services while ensuring digital sovereignty of service customers. In this chapter, motivated by the *privacy by design* principle, we apply our model-based privacy analysis methodology, proposed in Chapter 4, to the IDS. The approach is supported by the CARiSMA tool.

8.1 Introduction

Privacy has recently become a major factor in any kind of software development [161]. Nowadays, most of the organizations that provide IT services require the personal information of their service customers, to perform their business processes. As a result, an enormous amount of personal data is collected, stored, and shared all over the world [54]. Failure to protect such data by organizations affects the data providers (service customers) negatively and may harm the reputation of service providers (organizations) and cause emotional or financial damages.

In Chapter 4, we highlighted the need for addressing privacy from the early phases of system design. Moreover, we stated that ensuring privacy is particularly challenging in industrial ecosystems, where several data processors may process personal data. An example of such ecosystems is the *Industrial Data Space* (*IDS*) [18]. The IDS aims at establishing a network for trusted data exchange between different organizations, which provide or process data. A strategic requirement of the IDS is to provide secure data supply chains to ensure a high level of confidence when exchanging and processing data. The current reference architecture of the IDS ([18]) does not consider privacy explicitly. In particular, it does not specify mechanisms to ensure that the principles on the processing of personal data introduced in Article 5 of the GDPR are respected.

In Chapter 4, we introduced a privacy analysis methodology. Our methodology generally enables one to verify whether the design of a system that processes personal data supports the privacy preferences. We use this methodology to perform a privacy analysis on the IDS to verify whether the privacy preferences of the data providers are supported. The reference architecture of the IDS [18] differs from the architecture analyzed in Chapter 4. In the IDS, the exchange of data is enabled through *connectors*, that is, dedicated communication servers for sending and receiving data. In this chapter, we make the following main contributions:

- We highlight the importance of addressing privacy of personal data in the reference architecture of the IDS (Section 8.2).
- We explain how PLAs (introduced in Chapter 3) may be established between data providers and data consumers to support the privacy analysis in the IDS (Section 8.3.1).
- We apply our model-based privacy analysis methodology to the IDS (Section 8.3.2).
- We validate our model-based privacy analysis with respect to the privacy targets (Section 8.4).

The remainder of this chapter is organized as follows. In Section 8.2, we describe the privacy challenges regarding the IDS. Section 8.3 demonstrates the application of our model-based privacy analysis to the IDS. In Section 8.4, we validate the privacy analysis applied to the IDS and provide a case study[1] demonstrating model-based risk analysis in the IDS. In Section 8.5, we investigate the related work. Section 8.6 concludes.

8.2 Addressing Privacy in the IDS

In the IDS, a *data provider* is a data controller, who exposes data (including personal data) to be exchanged in the IDS and specifies the privacy preferences of these data. In most cases, the data provider is identical with the *data subject*, who owns the data. Moreover, in the IDS, a *data consumer* either refers to a *data processor* who directly processes the provided data, or a *data controller* who transfers to other data processors the data and their privacy preferences.

The IDS initiative[2] was launched in Germany by representatives from business, politics, and research. The aim is to provide a virtual data space for secure data exchanges. Currently, the IDS includes 98 companies and organizations. The IDS establishes secure data supply chains from data source to data use while ensuring data sovereignty for data providers [18, 163]. It aims to provide a technology which is simple, reliable, and cheap for every citizen to use. In particular, the goal is to provide a platform for collaborative smart data analytics which supports true digital sovereignty of the private data of the citizens in order to put them in a sustainable position to control who receives their personal data and what they can do with it.

The main activities of the IDS are:

- **Providing data** is enabled through the *Broker* service. The *Broker* service indexes the metadata that is provided by a data provider (data controller). The metadata describe the source of data and contain a set of policies on using the data.
- **Exchanging data** is initiated by a data consumer requesting data from a broker. The request and the exchange of data are enabled by the IDS *connectors* that are deployed on each organization.

[1] This case study is based on the material and results provided in a bachelor's thesis [138].

[2] International Data Spaces Association, `https://www.internationaldataspaces.org/` (accessed: 2019-06-01)

- **Data Processing** is performed by the data applications and organizations' services.

A strategic requirement of the IDS is to ensure a high level of confidence during data exchange. To this end, the IDS reference architecture requires the use of a security profile in order to implement appropriate mechanisms to ensure secure data communication between connectors, provide proper access control mechanisms to support identity and access management and make use of cryptographic methods to establish trust across the entire business ecosystem and protect the IDS participants from fraud.

Article 5 of the GDPR stipulates six principles for the processing of personal data: personal data must be (a) processed lawfully, fairly and in a transparent manner in relation to the data subject, (b) collected for specified and legitimate *purpose*s, (c) adequate and limited to what is necessary regarding the purposes (*purpose*), (d) accurate and kept up to date (*granularity*), (e) kept in a form which permits identification of data subjects for no longer than is necessary for the purposes for which the personal data are processed (*visibility, and retention*), and (f) processed in a manner that ensures appropriate security of the personal data. The current security profile of the IDS does not require the use of mechanisms to ensure that the personal data processing in the IDS respects these principles—except the security principle (f). For instance, there is currently no mechanism prescribed to be used to ensure that personal data is only processed for a certain set of processing purposes or the stored personal data in a database of a data consumer are eventually deleted or restricted, or during personal data exchange, the granularity levels are respected.

The usage scenarios of the IDS span a large variety of domains, including automotive engineering, facility management, healthcare, and smart cities [114]. To illustrate the need for privacy in the IDS, consider the following concrete usage scenarios.

Sensors embedded in *car seats*: Such sensors are designed to improve the ergonomics of a smart car. The data produced by these sensors are transmitted to a central monitoring systems and stored in different databases. Such data may reveal physiological aspects of a car driver (for instance, by transmitting her/his weight average). Figure 8.1 illustrates an excerpt from the IDS system layer. The IDS virtual data space is demonstrated as a blue box. The organizations may exchange data through the connectors. Connectors are communication servers for sending and receiving data. In each organization, data (including personal data) are processed by applications that are deployed on each connector. These applications are either downloaded from the *App Store* of the IDS or are self-developed apps. The *telemetry data* sent by the sensors in a car, may be processed directly in the *car manufacturer*. However such telemetry data may be sent to an *insurance* company.

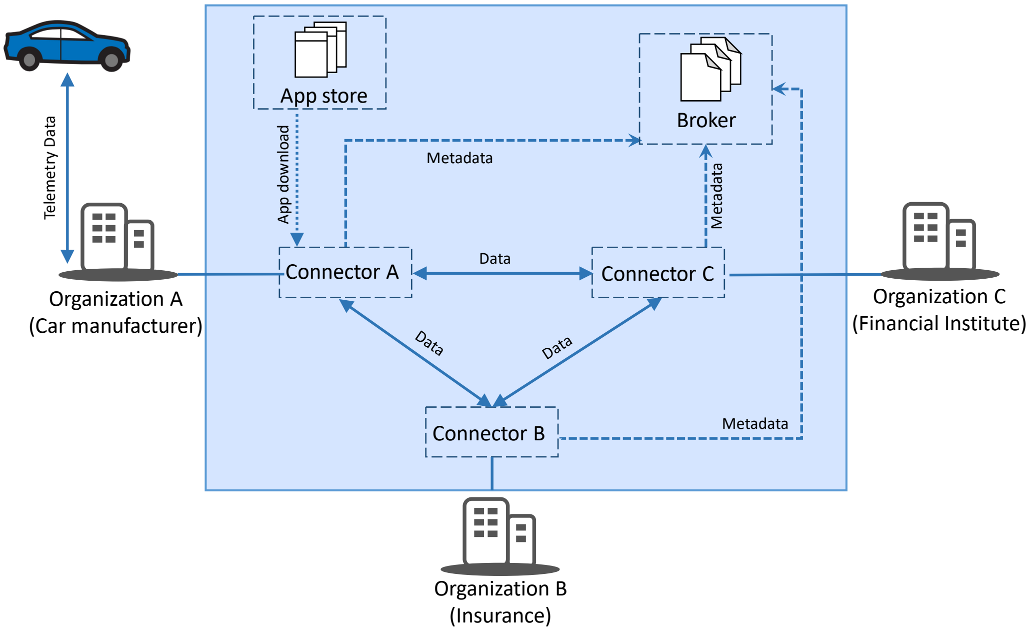


Figure 8.1: An illustration of the IDS system layer, including three organizations

Sensors embedded into infrastructural objects (for instance *trash cans*) to support smart services in smart cities: These trash cans may be managed by different operatives. The sensors embedded in these trash cans may log information about the *who* and *when* of trash can uses, and through the IDS connectors transmit such logged information. Such information may reveal the time schedule of the operatives. For instance when the operatives work or have breaks.

According to these two scenarios, the data that are exchanged between connectors may include some information about individuals. This makes it necessary to analyze the system's design of the connectors (as the central functional entity of the IDS) to verify whether the principles on the processing of personal data are supported. In the following section, we apply our model-based privacy analysis methodology to the IDS in order to ensure privacy protection in the early phases of system development.

8.3 Model-based Privacy Analysis for the IDS

We first describe how privacy preferences are specified for a piece of personal data. Afterwards, we apply our model-based privacy analysis to the IDS. To fully support the reference architecture of the IDS, a new privacy check is introduced.

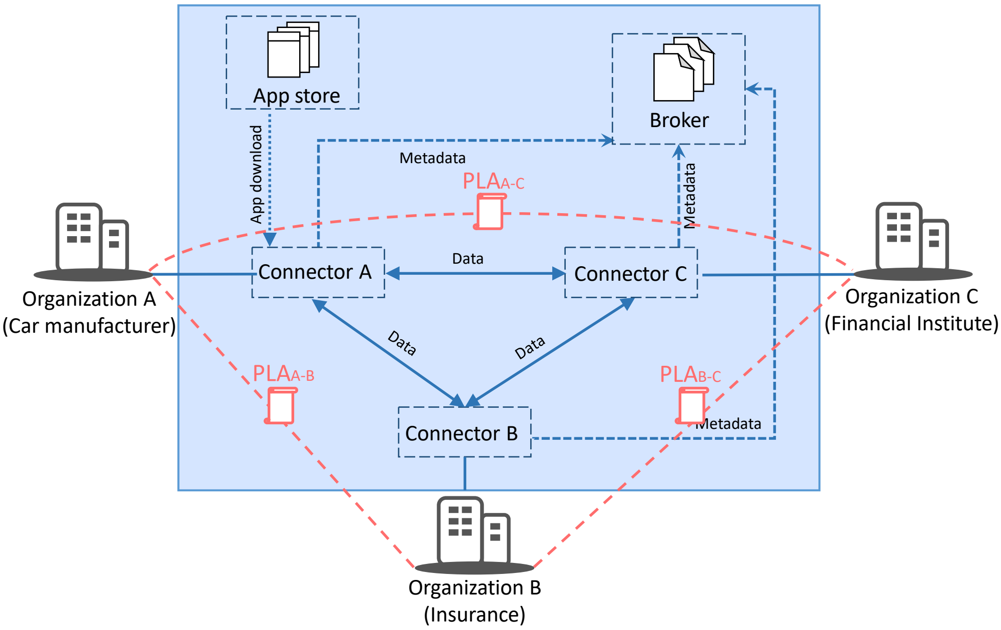


Figure 8.2: An illustration of the IDS system layer, including PLAs

8.3.1 Privacy Preferences

The security profile of the IDS manifests some high-level attributes such as hardware security, access controls, and authentication level [18]. To support privacy principles, the security profile of the IDS has to specify the personal data that are processed in the IDS. Moreover, a set of preferences on the processing of personal data in the IDS has to be defined. We use the definition of the privacy preferences provided in Section 3.3. The preferences are based on the four fundamental privacy elements introduced in [22], namely *purpose*, *visibility*, *granularity*, and *retention*.

In the IDS, we specify the privacy preferences for each piece of personal data in PLAs—see Sections 3.4 and 7.2.3 on PLAs. Between each two organizations that exchange data in the IDS a PLA is concluded. Additionally, in a PLA some specific information on each organization such as the organization's identity and the representative(s) are included. Figure 8.2 illustrates the excerpt from the IDS system layer, including three organizations and the concluded PLAs between them. The personal data processing in each organization has to support the privacy preferences included in PLAs. In the following section, we describe how our privacy analysis methodology is applied to the IDS to verify whether the privacy preferences are supported.

8.3.2 Privacy Analysis

According to the reference model of the IDS, to ensure privacy of personal data, data processing should be performed as close as possible to the data source, rather than be delegated to other organizations. If the data (including personal data) are intended to be transferred to external organizations, the data processing on an external organization must respect the privacy preferences specified in the PLA concluded between the two organizations (the data provider and the data consumer). To verify whether the privacy preferences are supported in this case, the system design of the organization, to which personal data are sent, has to be analyzed.

Connectors are the central functional entity of the IDS for exchanging and processing data. Independent of the apps being deployed on the connectors, a system model including several UML diagrams (in particular, *class, activity, component* and *deployment* diagrams) describes the structure and behavior of a connector. Such a system model belongs to the *configuration model* of a connector. According to the IDS, a configuration model describes the configuration of a connector in a technology-independent manner. Concerning the existing system models of connectors that are specified using UML[3], they are amenable to our proposed model-based privacy analysis methodology introduced in Chapter 4.

In Section 4.3.4, four privacy checks are introduced to analyze a system model concerning the four key privacy elements, namely *purpose*, *visibility*, *granularity*, and *retention*, to verify whether the privacy preferences are supported. To perform a privacy analysis on a system model, the system model has to be annotated with privacy elements. We use our proposed privacy extensions introduced in Section 4.3.3.

Based on the usage scenario of the embedded sensors in car seats, in Figure 8.3, a design model excerpt is provided. The activity diagram models the process of receiving and storing the weight of a car driver by a monitoring system. The weight is further transferred to a *research center* for further research. The data received from the sensors reveal physiological aspects of a car driver. Following Article 9 of the GDPR, such data belong to the special categories of personal data. Therefore, the object node is annotated with «`sensitiveData`». The *verifyWeight* action in the activity diagram is annotated with the stereotype «`recipient`» `{organization=reCent}` specifying that this action corresponds to an operation which belongs to the system model of the *research center*.

The operations in the classes are annotated with the stereotype «`objective`» and the relevant tags to express the processing purposes of each operation. The parameter of the operation *sendToReCent* is annotated with «`granularity`»

[3] Several UML system models derived from the concepts of IDS are discussed in [92, 138, 205].

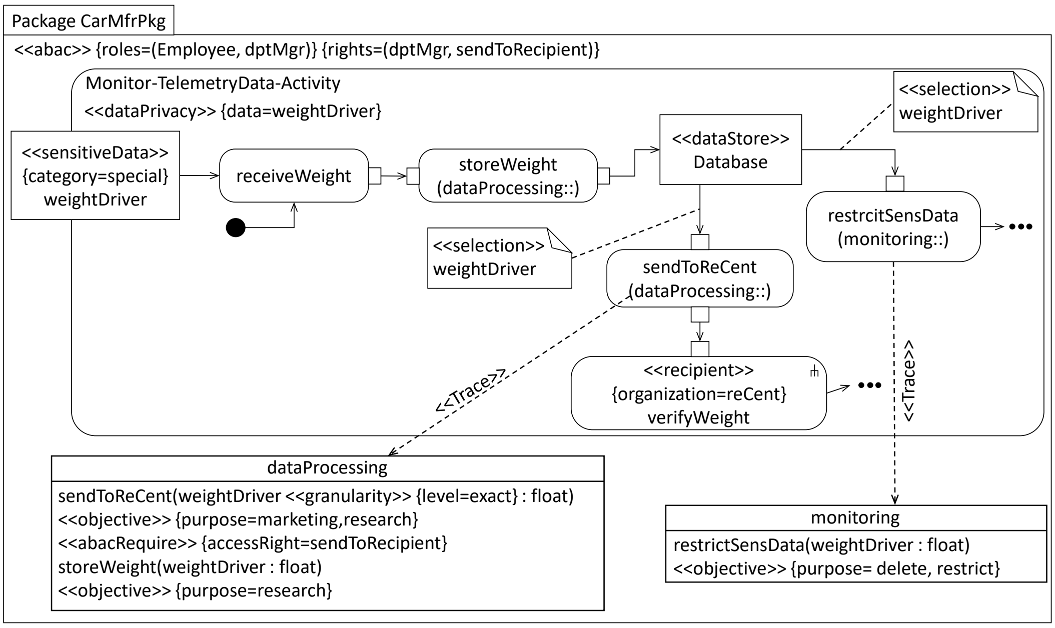

Package CarMfrPkg
<<abac>> {roles=(Employee, dptMgr)} {rights=(dptMgr, sendToRecipient)}
Monitor-TelemetryData-Activity
<<dataPrivacy>> {data=weightDriver}
<<sensitiveData>> {category=special} weightDriver
receiveWeight
storeWeight (dataProcessing::)
<<dataStore>> Database
<<selection>> weightDriver
restrcitSensData (monitoring::)
<<selection>> weightDriver
sendToReCent (dataProcessing::)
<<recipient>> {organization=reCent} verifyWeight
<<Trace>>
<<Trace>>
dataProcessing
sendToReCent(weightDriver <<granularity>> {level=exact} : float)
<<objective>> {purpose=marketing,research}
<<abacRequire>> {accessRight=sendToRecipient}
storeWeight(weightDriver : float)
<<objective>> {purpose=research}
monitoring
restrictSensData(weightDriver : float)
<<objective>> {purpose= delete, restrict}

Figure 8.3: Design model excerpt annotated with privacy profile

`{level=exact}`, specifying that the required precision level of the piece of personal data to be transferred to a recipient is *exact*. The stereotype «`abacRequire`» specifies the access rights of an operation. Using the access rights and concerning the stereotype «`abac`», the subjects who process a piece of personal data are identified. For instance, according to Figure 8.3, the *department manager* (*dptMgr*) process (send) the weight of a car driver to a recipient.

Consider the following privacy preferences for the weight of a car driver (*wcd*):

$$PRP_{wcd} = \{(research \mapsto (dptMgr, partial, 1M))\}$$

PRP_{wcd} specifies that the weight of a car driver may be processed by the department manager (*dptMgr*) for the purpose of *research* for the period of *one month* with the precision level *partial*. Consider that the *research* purpose does not subsume the *marketing* purpose.

According to our proposed privacy checks (Section 4.3.4), to analyze the model provided in Figure 8.3, first the objectives of the operations, annotated in the system model, are verified with respect to the authorized purposes specified in the privacy preferences. Particularly, two actions process the weight of a car driver: *sendToReCent* and *storeWeight*. Concerning the corresponding operations in the class *dataProcessing*, *sendToReCent* operation processes the weight of a driver for two purposes: *research* and *marketing*. However, following the privacy preferences, *wcd* is only authorized to be processed for the purpose of *research*; this is a privacy design violation. A further privacy design violation, which is included in the analysis results—following performing a privacy analysis—is related to the granularity level. The *sendToReCent* operation requires the *exact* precision level to process *wcd*, however, according to the PRP_{wcd}, *wcd* is only authorized to be processed with *partial* precision level.

Broker-check: According to the system layer of the IDS in Figure 8.1, an organization through its connectors may exchange data with an IDS broker. Such a data exchange is enabled through metadata, which describe the source of data and provide a set of policies on using the data. A data exchange with an IDS broker should not contain personal data. Metadata only aim to initiate data exchanges between IDS connectors.

We propose a new simple privacy check (*broker-check*) to ensure that a data exchange between an organization and an IDS broker does not include personal data. Given an activity diagram which models the data exchange with an IDS broker, the metadata that are stored in a *namedElement*, such as a *dataStore* node, annotated with «`recipient`» {`organization=IDSbroker`} should not be annotated with «`sensitiveData`». Currently, to check this, we need to verify whether the parameter of an operation that stores data in a database of a broker is annotated with «`sensitiveData`».

Figure 8.4 shows an excerpt from an activity diagram specifying the process of storing metadata in the database of an IDS broker. The *dataStore* node is annotated with «`recipient`» {`organization=IDSbroker`} specifying that this node is a database in an IDS broker. The object which is stored in this *dataStore* is annotated with «`sensitiveData`». This is a privacy design violation, identified by the broker-check.

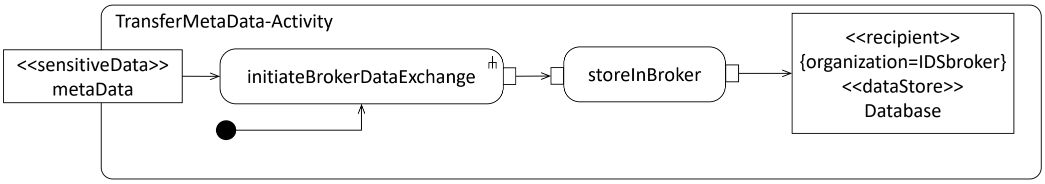


Figure 8.4: Design model excerpt

The *Broker-check* is, in fact, a very simple check that investigates which operation stores (for instance concerning a processing purpose *store*) a piece of personal data annotated with the stereotype «`sensitiveData`» in a certain database (a broker's database). To fully support such a check, a privacy analysis at runtime is required that enables one to analyze the source code of a connecter. Such an analysis at runtime is defined as a future research direction for this thesis (Section 9.3.1).

8.4 Discussion and Results

In this section, we first validate our privacy analysis methodology applied to the IDS, concerning how effectively the privacy targets (Section 5.2.3) are addressed. Afterwards, we introduce a case study derived from the usage scenarios of the IDS [114], showing how to support a security analysis using model-based risk analysis.

8.4.1 A Validation of the Model-Based Privacy Analysis Concerning Privacy Targets

Since the system models of the IDS have to be treated in confidence, we do not provide the actual system models of the IDS. The scenarios and the design model excerpts presented in this chapter are based on the existing system models (UML) and example scenarios of the IDS [114]. Generally, by applying the model-based privacy analysis to the IDS (the car seat's sensors scenario), we noticed that such an analysis can successfully support *privacy by design* (*PbD*) in the IDS. The identification of design violations, which specify that a system model is not fully in compliance with a set of privacy preferences, assists practitioners to support privacy requirements in the early phases of system development and facilitates the integration of privacy enhancing technologies (PETs) into the system design.

Particularly:

- We described the importance of addressing the privacy of personal data in the reference architecture of the IDS. For this, we described two example scenarios from the IDS, in which failures to ensure privacy protection may affect the data providers and the data consumers.
- We leveraged the privacy preferences and the PLAs introduced in Chapter 3 to support the privacy of personal data in the IDS.
- We applied the model-based privacy analysis introduced in Chapter 4 to support the *PbD* principle in the IDS.

The privacy analysis is supported by CARiSMA (See Section 7.2).

As previously mentioned (Section 5.2.3), in [158], the authors provide a systematic support for representing privacy requirements in the form of privacy targets. The privacy targets are derived from legal privacy and data protection principles. We proposed to add three new privacy targets to the list of existing privacy targets (Section 5.3.4) To validate our proposed model-based privacy analysis, we verify how by applying our privacy analysis methodology to the IDS, the privacy targets are addressed.

The IDS reference architecture and it's security profile support a number of privacy targets. For instance, *accountability*, *security of data*, and *data accuracy and integrity* are supported by the security profile of the IDS. Moreover, the IDS provides appropriate mechanisms to ensure *limited storage*, *data portability*, and *notifications to the third party* [18].

The IDS does not prescribe mechanisms to support the privacy targets that are related to the privacy elements, namely *purpose*, *visibility*, *granularity*, *retention*. Our privacy analysis provides a mechanism to analyze a system model of the IDS to verify whether the privacy elements, as well as the relevant privacy targets, are supported by a system. For instance, the specification of authorized purposes in a PLA and their comparison with the processing purposes of a system, support the privacy targets *P1.2 - P1.4*, *P1.6*, *P3.1*, and *P5.2*.

8.4.2 Supporting Security and Privacy Analysis by Model-Based Risk Analysis in the IDS

In the context of a Bachelor's thesis [138][4], our model-based privacy analysis (particularly the visibility check), the UMLsec security analysis and our risk analysis methodology—the risk analysis is the basis of the *privacy impact assessment* methodology introduced in Chapter 5—are applied to a case study derived from the usage scenarios of the IDS. The case study does not release the real models that are established in the context of the IDS project. The diagrams of this case study are based on the reference architecture of the IDS. The case study models the imaginary *Sunshine Weather Service* company. This company installs specific sensors in public to monitor the weather conditions.

In the context of this case study, first, the system is modeled using UML diagrams. Afterwards, the diagrams are annotated with the UMLsec annotations [128] and the *rabac* profile (see UML privacy extension in Section 4.3.3). Eventually, three security and privacy checks: *secure dependency*, *secure links*, and *visibility* checks are conducted on the system model.

[4] The author of this PhD thesis was one of the supervisors of the Bachelor's thesis.

Table 8.1: The privacy targets supported by the IDS and the privacy check (described in Chapter 4).

Privacy targets	**Supported by**
P1.1 Ensuring fair and lawful processing by transparency	PC
P1.2 Ensuring processing only for legitimate purposes	PC
P1.3 Providing purpose specification	PC
P1.4 Ensuring limited processing for specified purposes	PC
P1.5 Ensuring data avoidance	IDS
P1.6 Ensuring data minimization	PC
P1.7 Ensuring data quality, accuracy and integrity	IDS
P1.8 Ensuring limited storage	IDS
P1.9 Ensuring the categorization of personal data	PC
P1.10 Ensuring the prevention of discriminatory effects on natural persons	PC
P2.1 Ensuring legitimacy of personal data processing	PC
P2.2 Ensuring legitimacy of sensitive personal data processing	PC
P3.1 Adequate information in case of direct collection of data	PC
P3.2 Adequate information where data is not obtained directly	IDS
P4.1 Facilitating the provision of information about processed data and purpose	PC
P4.2 Facilitating the rectification, erasure or blocking of data	PC
P4.3 Facilitating the portability of data	IDS
P4.4 Facilitating the notification to third parties about rectification, erasure and blocking of data	IDS
P5.1 Facilitating the objection to the processing of data	PC
P5.2 Facilitating the objection to direct marketing activities	PC
P5.3 Facilitating the objection to data-disclosure to others	PC
P5.4 Facilitating the objection to decisions on automated processing	IDS
P5.5 Facilitating the data subjects right to dispute the correctness of machine conclusions	IDS
P6.1 Ensuring the confidentiality, integrity, availability, and resilience	IDS
P6.2 Ensuring the detection of personal data breaches and their communication to data subjects	IDS, PC
P6.3 Ensuring the effectiveness of technical and organizational measures. A 32.1(d)	PC, IDS
P7.1 Ensuring the accountability	IDS

PC:	Our proposed privacy checks (Chapter 4).
IDS:	Supported by the IDS

Table 8.2: The table demonstrates to which threats of the *Treacherous 12*, the results of the three checks of CARiSMA refer [138].

Treacherous 12 threats [50]	Checks
Data Breaches	Secure links, Secure dependency, Visibility check
Weak Identity, Credential and Access Management	Secure dependency, Visibility check
Insecure APIs	Secure dependency
System and Application Vulnerabilities	Secure links
Account Hijacking	Secure links, Secure dependency
Malicious Insiders	Secure links, Visibility check
Advanced Persistent Threats (APTs)	Secure links, Secure dependency, Visibility check
Data Loss	Secure links, Visibility check
Insufficient Due Diligence	Secure links
Abuse and Nefarious Use of Cloud Services	-
Denial of Service	Visibility check
Shared Technology Issues	Secure links, Secure dependency, Visibility check

The results of the analysis indicate a set of design violations in the system design. Since the sensors in the *Sunshine Weather Service* company do not exchange personal data, the identified violations are only security relevant—precisely no privacy design violation is identified. For instance, an identified violation indicates that a communication link between a sensor and the central database of the *Sunshine Weather Service* company is not secure and an attacker can read the exchanged data between the two devices (a sensor and the database). This design violation was identified by performing the *secure links* check [128] on the deployment diagram that model the relationship between physical elements (devices) of the *Sunshine Weather Service* company.

To validate the results of the conducted analysis, the author of the Bachelor's thesis verified which threats may be identified from the analysis results. This validation leverages our concepts proposed in Section 5.3.3 to identify harmful activities and threats in an impact assessment. As mentioned in Section 5.2.4, several documents are available to introduce threats. In the Bachelor's thesis, the *Treacherous 12* [50] document (cloud computing top threats) is used as a source of threats. Table 8.2 demonstrates how the threats can be respectively identified by the checks.

The results of the case study (particularly Table 8.2) state that the three checks of CARiSMA may be used to identify a large proportion of the *Treacherous 12* threats and therefore, it provides a basis to conduct a model-based risk analysis. However, CARiSMA includes several other checks which can be used to enhance and refine the case study in the future.

8.5 Related Work

In this section, we provide the earlier work on the application of privacy analysis approaches to specific case studies. Further related works on different privacy analysis approaches are provided in Section 4.6.

In [62], Joyee De et al. describe the notions of *harm*, *feared events*, *privacy weakness*, and *risk sources*. They further provide a relationship among these notions using suitable examples within the smart grid systems. This work provides a promising foundation to conduct a risk assessment process for smart gird systems. In their work, they do not specify how the privacy harms and weaknesses may be identified by analyzing the design of the smart grid systems.

In [95], Guerriero et al. provide a prototype tool to enhance data-intensive applications with attribute-based access control policies, propagate such policies using model-driven pipeline and monitor their validity at runtime. Their approach is orthogonal to our approach and provides a promising technique to enhance a system model (UML component diagram) with access control mechanisms. In our approach, we first identify privacy design violations by analyzing the behavior (activity diagram) and the structure (class diagram) of a system.. We further provide a mechanism to enhance the behavior and the structure of a system with several privacy enhancing technologies (including access control mechanisms) and patterns.

In [112], Hu et al. proposed a novel methodology to detect and resolve privacy conflicts in collaborative data sharing within online social networks. In their conflict resolution mechanism, they attempt to find an optimal tradeoff between privacy protection and data sharing. They do not consider the system design of an online social network to identify the privacy conflicts.

In [29] and it's recent variation [110], the authors developed a framework for privacy-aware design in the field of ubiquitous computing. In this framework, a set of questions is provided that enables the designers to evaluate a system. Although these frameworks are fast to implement and inexpensive, they only identify a set of static privacy problems in systems, and no privacy analysis on the system's design is performed.

In the context of a Master's thesis [92], a catalog of security requirements for a data exchange between two sample connectors in the IDS is provided. This catalog is the result of performing an asset-driven threat analysis from an attacker's perspective. This threat analysis determines 67 different attack scenarios, from which 27 security-critical threats are identified, and rendered into security requirements. Furthermore, the identified security requirements are assigned to a set of controls derived substantially from the BSI [39]. This work particularly resembles our *privacy impact assessment* methodology introduced in Chapter 5. The author of the Master's thesis stated that the privacy and data protection principles (the principles prescribed in Article 5 of the GDPR on processing personal data) are out of scope of the Master's thesis, and a reference to our work introduced in this chapter is given. Our model-based privacy by design methodology introduced in this PhD thesis is a general methodology, and as demonstrated in this chapter, it can be applied to the IDS. In the future, using the privacy targets (Section 5.2.4) and the Solove's harmful activities (Section 5.2.4), the catalog of requirements provided in the Master's thesis [92] has to be extended.

8.6 Preliminary Conclusion

We explained the importance of addressing privacy in the Industrial Data Space (IDS). We applied our model-based privacy analysis methodology (introduced in Chapter 4) to the IDS. To support the privacy of the data exchange between an IDS broker and a connector we proposed to extend our privacy analysis by a new privacy check. Several system models derived from the usage scenarios of the IDS are used to evaluate our privacy analysis methodology. We discussed the results of the application of the model-based privacy analysis to the IDS concerning the privacy targets that are derived from the GDPR.

Chapter 9

Conclusions, Limitations and Outlook

In this chapter, we outline the key conclusions of this thesis. Moreover, we present a selection of assumptions and limitations for our model-based *privacy by design* methodology. Finally, we describe the directions for future work.

9.1 Conclusion

Privacy by design (*PbD*) calls for considering privacy concerns in the design of IT systems from the early phases of the system development. However, *PbD* cannot simply be achieved by integrating a set of privacy controls into a system. We identified four challenges to operationalize *PbD*: What are the privacy concerns, and how can such concerns be identified? How is it possible to verify whether the privacy concerns in a system are properly supported? What is at risk in a system when processing personal data, and how can the risks be identified? Which controls and measures can adequately mitigate those risks?

We introduced a model-based methodology to operationalize *PbD*. This methodology assists a system developer to consider privacy from the onset of the system development through integrating appropriate privacy controls into the system design. Our methodology relies on the definition of privacy preferences and comprises three sub-methodologies.

To embed privacy into a system model, a developer first needs to identify where privacy is needed in the system model. This calls for a system model analysis. An analysis has to verify whether a system model supports a set of privacy preferences. Therefore, identifying privacy preferences is a necessary step toward operationalizing *PbD*. We provided a definition for the privacy preferences. The four key privacy elements, namely *purpose*, *visibility*, *granularity*, and *retention*, constitute the privacy preferences. In Chapter 3, we discussed how the lattice structures that we use to define privacy preferences, enable one to efficiently express the privacy preferences. Moreover, our definition adheres to the principles relating to the processing of personal data prescribed by the GDPR (Article 5), where purpose is the central privacy element and the other three elements are defined in regard to the purpose.

We further introduced privacy level agreements (PLAs) to establish agreements between data controllers and data processors, specifying the privacy preferences of the personal data. We argued that a piece of personal data may be processed by several data processors. This demands a systematic mechanism to specify the privacy preferences of personal data. Since the original PLA outline introduced by Cloud Security Alliance is heavily based on the former data protection regulation of the EU, we updated the PLA outline with respect to the GDPR.

We introduced a modular, formally grounded model-based privacy analysis methodology. A privacy analysis verifies whether a system model supports a set of privacy preferences. Our proposed privacy analysis comprises four privacy checks, which adhere to the four key privacy elements. The results of an analysis denote a set of privacy design violations. Such violations determine the need for integrating privacy controls into a system design.

We described that in today's digital society, a data processor may depend on other processors to process a piece of personal data—personal data processing in an industrial ecosystem. In this case, to verify whether the personal data processing is authorized, the system models of several organizations have to be analyzed. Thus, a modular privacy analysis is required that analyzes the system design of each organization separately. Such a modular analysis is particularly beneficial when an existing system model is modified, or a new data processor is added to an industrial ecosystem. In such cases, a complete analysis of the ecosystem is not necessary.

Our proposed privacy analysis methodology was applied to three practical system models that were provided by the industry partners of the VisiOn project (an EU research project that we participated in). We provided elaborated tool support (an extension of CARiSMA) to enable a privacy analysis. Our observations and the industry partners' reports indicated that our model-based privacy analysis methodology successfully assisted a system developer in identifying the privacy design violations in a system model.

Using our observations and the results of a survey performed by us to obtain the expertise of the industry partners, we investigated the support required by the industry partners to perform our proposed model-based privacy analysis. The results indicated that although initially system modeling and conducting a privacy analysis by CARiSMA were not easy for the industry partners, our substantial support including training, workshops, webinars, and manuals allowed the partners to model their systems and perform an analysis.

We introduced a novel model-based privacy impact assessment (*PIA*) methodology. In a *PIA*, the results of our privacy analysis are evaluated to identify the privacy risks. We used the list of privacy targets introduced by BSI (Federal Office of Information Security in Germany) to assess the risks. Our risk assessment relies on the criticality of personal data that is processed and the protection demands (with respect to the privacy targets) of both data controllers and data processors (the key stakeholders when processing personal data).

After analyzing the three system models, on each system model, a *PIA* is conducted to identify privacy risks. We could successfully identify a set of privacy risks in each system model. Moreover, the results of a comparative evaluation showed that our *PIA* methodology particularly supports three *PIA* legal guidelines [59, 86, 159] with a thorough description of information-flows, concrete risk assessment and specific plans to implement controls.

Our proposed *PIA* methodology suggests a set of privacy controls to mitigate the identified risks. However, choosing appropriate controls and integrating them into a system model are complex tasks and involve several issues. The privacy controls are abstract in nature and cannot be directly applied to a system model. Moreover, the interrelations between the controls and their costs have to be considered. In Chapter 6, we proposed a methodology to perform a comprehensive privacy enhancement of a system model concerning all these issues. Due to the abstract nature of the privacy controls, we mapped them to a set of privacy features including privacy design strategies, privacy design patterns, and privacy-enhancing technologies. Furthermore, we performed an investigation of the interactions between the features and captured the respective interrelations and dependencies in a feature model. To estimate the cost of the privacy features, we proposed a novel model-based cost estimation approach by customizing functional point analysis for applying to activity diagrams (which were used to model the behavior of a system in this thesis).

Following a privacy analysis and a privacy impact assessment, the privacy enhancement of a system model is the final step toward operationalizing *PbD*. A privacy enhanced system model may be analyzed iteratively to verify whether the violations and the arising risks are mitigated. This gives rise to the fact that our methodology may be applied to an existing system to verify whether a privacy enhancement of the analyzed system is required.

9.2 Assumptions and Limitations

The assumptions and limitations of our model-based methodology: Our methodology introduced in this thesis to operationalize *privacy by design* is a model-based methodology. System models (UML class and activity diagrams) are required to conduct a privacy analysis as well as a privacy impact assessment. Our system model analysis only identifies the potential design violations in a system model with respect to the privacy preferences. This gives rise to the facts that there is no assurance that all privacy issues of a system will be identified and the real implementation of a system guarantees all privacy preferences. Concerning these facts and since the identification of the risks in this thesis relies on the results of our model-based privacy analysis, our *PIA* is not capable of identifying all privacy risks of a system. However, based on our outlined conclusion, system model analysis covers the early phases of a system development and identifies a set of privacy threats and risks which facilitate the privacy enhancement of a system model.

Furthermore, the identification of certain violations and associated threats in a system is only possible at runtime by means of analyzing the real implementation of the system. For instance, our privacy analysis comprises four privacy checks, including the *retention* check. Given a system model that specifies the design of a system, the *retention* check only investigates whether an appropriate mechanism is available to remove or to restrict personal data. However, concerning the design of a system in the early phases of development, it is not possible to verify whether personal data will be eventually removed or restricted.

Our validations based on the VisiOn case studies: To evaluate the applicability of the three sub-methodologies introduced in this thesis, we applied them to three system models established during the VisiOn project. The three system models are provided by our industry partners (three public administrations). The public administrations (PAs) further expressed their privacy concerns in their system models with an annotation mechanism provided by us. Due to their lack of knowledge to model their systems and to annotate the resulting system models, we substantially supported the PAs to produce the three annotated system models. Thus, the system models that are used for evaluating our concepts may also reflect our knowledge. Moreover, the system models include a limited number of diagrams and particularly, by the enhancement of a system model, we focused only on enhancing activity diagrams.

A consistent terminology for the lattices: Specifying the privacy preferences and annotating the system models are necessary to perform a privacy analysis. The privacy preferences of personal data are structured in lattice structures. We saw that in an industrial ecosystem, a piece of personal data may be processed by several data processors. In this case, the system model of each service provider is analyzed to verify whether privacy preferences are supported. This calls for ensuring consistency in lattices and system models' annotations. In other words, the system models and the sets of all possible purposes, subjects (who may process personal data), the granularity levels, and retention conditions have to follow a consistent terminology.

Assumption on certain predefined categories, values, and ordinal scales: In this thesis, we introduced several categorizations and ordinal scales. For instance, we defined four categories of personal data in Section 4.3.3.1 and following this, in Section 5.3.4 we assigned certain values to these categories to assess the privacy targets at risk and identify privacy risks. Moreover, a *PIA* calculates the final *impact assessment* scores of emerging risks relying on an ordinal scale (Section 5.4), which enables estimating the protection demands of data controllers and data processors for privacy targets. We further defined four ranges to categorize *impact assessment* scores. These ranges are necessary to suggest privacy controls, which are important artifacts to enhance system models.

Furthermore, our methodology needs a set of default (preexisting) values. After conducting a *PIA*, a set of privacy controls are suggested to mitigate the emerging risks. Having mapped the suggested privacy controls to a set of privacy features, an appropriate selection of features has to be chosen to enhance a system model. This selection relies on the strength of features to mitigate the risks and the costs of the features. The strength is denoted by four levels of rigour which have to be assigned to the features before conducting an enhancement. The costs are estimated by a model-based method which uses a set of predefined complexities for the model elements that are considered in a cost estimation.

Such predefined categorizations, values and ordinal scales only provide a baseline to show the operationalization of *PbD* in this thesis and can be extended or modified concerning various factors.

Avoiding likelihoods when assessing risks: The calculation of the *impact assessment* scores for privacy targets that are at risks is a necessary step toward suggesting privacy controls to enhance a system model. The *impact assessment* scores rely on the severities of the violations identified after performing a privacy analysis. The likelihood that a threat may occur is ignored. In Section 5.3.4, we justified this assumption.

9.3 Outlook

Concerning our outlined conclusion, our model-based privacy by design methodology provides a step forward toward fulfilling Article 25 of the GDPR on *privacy by design*. Still, a variety of research directions have to be explored in the future.

9.3.1 Privacy Analysis at Runtime

The system model privacy analysis that we introduced in this thesis concerns the design of a system in the early phases of system development. We further elucidated that our methodology may be applied to an existing system (provided by an annotated system model), thereby identifying privacy risks and discovering the mandatory privacy enhancements. However, a number of privacy violations can be only determined at runtime. For instance, above (in Section 9.3), we stated that using the privacy check *retention*, it is not possible to verify whether personal data will be eventually removed or restricted. This, in fact, gives rise to a future direction to extend our proposed privacy analysis by bridging the gap between the design and the runtime phases of a system through synchronizing the annotated system models with source code. To achieve this, an appropriate source code annotating mechanism and a methodology to monitor the source code execution with respect to the source code annotations are required.

9.3.2 Model-Based Discrimination Analysis

In Chapter 5, a new privacy target is added to the list of existing privacy targets, to ensure the prevention of discriminatory effects on natural persons. Algorithmic decision-making systems are used to automatically make decisions in different industrial domains [132, 174]. To avoid discrimination against natural persons in such systems, the causes (instances) of discrimination have to be determined. Using our proposed privacy analysis methodology, we can verify whether a set of specific types of personal data (protected characteristics) such as *race* or *physical status* of a person are processed for certain purposes such as *profiling*. However, such an analysis only reveals that in a system, discrimination may occur. Ramadan et al. [174], introduced an approach toward a model based discrimination analysis. To explicitly identify the causes of discrimination in a system model, it has to be determined whether the decisions made by a system relies on a set of protected characteristics. This calls for an information-flow analysis concerning certain types of personal data. It has to be verified whether in a model (for instance a UML state diagram) that demonstrates the information-flow in a system, upon changing an input (that is related to a protected characteristic) of an event, the final decision (final state) will also be changed.

9.3.3 Investigating the Means to Support Performing a Model-Based Privacy Analysis

The methodology introduced in this thesis to operationalize *PbD* requires expertise in system modeling with UML. Furthermore, before performing a privacy analysis, the system models have to be annotated with the UML profiles introduced in this thesis. In Chapter 4, we investigated the support and the expertise required by the users of our privacy analysis methodology. We conducted a survey to obtain the expertise of the industry partners of the VisiOn project. This study (investigation) is the first step toward future research in this area. Our research conducted toward identifying the required support to perform a model based *PbD* can be extended in several directions including (**I**) increasing the population of the survey, (**II**) conducting an objective assessment rather than a subjective one, (**III**) evaluating the usefulness of our proposed concepts.

9.3.4 Refined Cost Estimation

Exploring the trade-offs between the costs of the privacy features and the arising risks, using model-based software engineering is an emerging research direction. Our model-based cost estimation approach introduced in Chapter 6 is a step forward toward this research direction. Our approach relies on counting certain elements in data flow views of reusable aspect models (RAMs). We extended RAMs by data flow views which are modeled by activity diagrams. The cost estimation approach can be extended in two directions: (**I**) defining more rigorous weights for the elements that are involved in the cost estimation approach. Currently only a set of predefined weights are used to estimate the costs, however, these weights can be refined by a deeper analogical analysis based on the experiences of the privacy experts, or by learning from historical data. (**II**) Taking into account all the views of RAMs to estimate the costs of the privacy features (modeled with RAMs). RAMs originally comprise three views, namely structure (modeled with class diagrams), message (modeled with sequence diagrams), and state (modeled with state diagrams) views.

Appendix A

Glossary

Accountability: The security goal that generates the requirement for actions of an entity to be traced uniquely to that entity [1].

Authentication: Verifying the identity of a user, process, or device, often as a prerequisite to allowing access to resources in an information system[1].

Authorization: Access privileges granted to a user, program, process, or the act of granting those privileges[1].

Availability: Ensuring timely and reliable access to and use of information[1].

Confidentiality: Preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information[1].

Data controller: A data controller determines the purposes and the means for the processing of personal data [198].

Data processor: A data processor processes personal data on behalf of the controller [198].

Enterprise: An enterprise means a natural or legal person engaged in an economic activity, irrespective of its legal form, including partnerships or associations regularly engaged in an economic activity [198].

General identification number: An identifier of general application such a national identification number [198]. The *Social Security Number* (*SSN*), which is introduced in this thesis, is a general identification number.

[1] The online NIST glossary of key information security terms `https://csrc.nist.gov/glossary` (accessed: 2019-06-01)

Industrial ecosystem: An environment where a group of enterprises are engaged in the processing of personal data.

Integrity: The property that sensitive data has not been modified or deleted in an unauthorized and undetected manner[2].

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 [198].

Privacy design violation: The result of a privacy analysis, which refers to a specific behavioral or structural aspect of a system model, which does not respect the privacy preferences—specified regarding the four key privacy elements.

Privacy Level Agreement (PLA): A PLA is intended to be an appendix to a service level agreement (SLA), and to describe the level of personal data protection provided by service providers to service customers in a structured way [49].

Privacy preferences: The purposes and the means of the processing of personal data. In this thesis, the privacy preferences are defined based on the four key privacy elements: purpose, visibility, granularity, and retention.

Privacy-relevant data: The data that initially are not considered as personal data, however later risks for the privacy of individuals based on such data may become apparent [58].

Processing: Any operation performed on personal data such as collection, recording, organization, structuring, storage, adaptation or alteration, retrieval, consultation, use, disclosure by transmission, dissemination or otherwise making available, alignment or combination, restriction, erasure or destruction [198].

Recipient: A recipient is a natural or legal person, public authority, agency or another body, to which the personal data are disclosed. Public authorities which may receive personal data in the framework of a particular inquiry shall not be regarded as recipients [198].

Risk: A combination of the likelihood of an event and its consequence [120].

[2] The online NIST glossary of key information security terms `https://csrc.nist.gov/glossary` (accessed: 2019-06-01)

Risk analysis: A systematic use of information to identify and to estimate risk [120].

Service customer: In this thesis, a service customer is either a data processor, who directly processes the provided data, or a data controller, who transfers to other data processors the data and their privacy preferences.

Service provider: In this thesis, a service customer is a data controller, who provides personal data and specifies the privacy preferences of these data.

Social Security Number (SSN): The *SSN* (*AMKA*[3] in Greek) is the insurance ID of a person in Greece.

Special categories of personal data: Personal data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, or trade union membership, and the processing of genetic data, biometric data for the purpose of uniquely identifying a natural person, data concerning health or data concerning a natural person's sex life or sexual orientation [198].

Subject: In this thesis, we use the term *subject* for various data users, including a natural person, a department, an organization, or any resource that may process data.

Threat: A potential cause of an unwanted incident, which may result in harm to a system or organization [120].

[3] `http://www.amka.gr/tieinai_en.html` (accessed: 2019-06-01)

Appendix B

A Comparison Between the GDPR and the Directive 95/46/EC

In Section 3.4.1, a brief description of the differences between the GDPR and Directive 95/46/EC is provided. The GDPR repeals Directive 95/46/EC, thereby updating and modernizing the principles stated in the Directive 95/46/EC to guarantee privacy rights.

In Table B.1, an excerpt of the comparison between the GDPR, and Directive 95/46/EC is provided. The complete comparison is documented in an Excel file[1].

In the first column of Table B.1, the GDPR articles [198] are listed. In this column the title of each article is shortened. For each article in the first column, in the second column the relevant (similar) article(s) of Directive 95/46/EC [197] is (are) provided. Furthermore, in the second column, we shortly describe the differences in regard to the GDPR.

[1] https://cloud.uni-koblenz-landau.de/s/ocRXY9nJqDWzgpA (accessed: 2019-06-01)

Table B.1: The differences between the GDPR and Directive 95/46/EC

GDPR	Directive 95/46/EC, and the results of the comparison
Article 4 Definitions	New definitions such as genetic and biometric data, restriction of processing, profiling, pseudonymization, consent, personal data breach are added
Article 5 Principles	Article 6 (in the GDPR, concepts such as accuracy, data minimization, purpose limitation, storage limitation, integrity and confidentiality, accountability are explicitly stated.)
Article 6 Lawfulness of processing	Article 7 (in the GDPR, the purpose limitation and specification is described in detail.).
Article 7 Conditions for consent	The conditions of consent are mentioned in Article 7 and 8 of the directive. In the GDPR, the conditions for consent have been strengthened.
Article 8 Conditions on child's consent	Not included
Article 9 Special categories of personal data	Article 8 (in the GDPR, special categories are extended with new categories such as genetic, and biometric data.)
Article 17 Right to erasure	Article 13
Article 18 Right to restriction	Article 13
Article 19 Notification obligation	-
Article 20 Right to data portability	-
Article 25 PbD	Not included as a separate article (following Article 25 of the GDPR, PbD is now legally binding).
Article 35 DPI	Not included as a separate article (following Article 35 of the GDPR, PIA is now legally binding).
Article 87 National identification number	Article 8 paragraph 7 (Not included as a separate article in the directive.)

Appendix C

The Updated PLA Outline

In Section 3.4.1, we highlighted several differences—including updated or newly added principles—between Directive 95/46/EC [197], and the GDPR [198]. Moreover, in Section 5.3.6, we suggested extending privacy level agreements (PLAs) to report the results of privacy impact assessments. Concerning the fact that the current PLA outline [49] is heavily based on Directive 95/46/EC, and to support the above-mentioned differences and suggestions, we propose an extension (update) of the current PLA outline. We only focus on the sections of the outline that have to be extended (updated), the rest remains unchanged.

2-Ways in which the data will be processed:

In the current PLA outline, this section provides details on **(I)** the purposes of the processing for which the data are intended and the necessary legal basis to carry out such processing as per Article 7 Directive 95/46/EC; **(II)** any further information such as recipients, the obligatory or voluntary nature of providing the requested data, and the existence of the right of access to and the right to rectify the data concerning the data subject.

According to our definition of privacy preferences introduced in Section 3.3, as well as Article 5 of the GDPR, we believe that in addition to providing the purposes of the processing, to each purpose particularly a set of subjects with authorized rights to process data (in regard to the authorized purpose), a granularity level, and a retention condition have to be assigned.

Moreover, as mentioned in Section 3.4.1, the GDPR prescribes a set of conditions for consent. Although Article 7(a) Directive 95/46/EC stipulates that the data subject has to unambiguously give his consent, the consent conditions are not necessarily supported by the current PLA outline. We adjust the PLA outline concerning the consent conditions, and propose to add a subsection that specifically specifies the personal data consent.

In this thesis, we introduced a categorization of personal data, which is essential to perform a privacy impact assessment. We believe that PLAs have to include a categorization of personal data. Therefore, we propose to include it in this section.

4-Ensuring Privacy by Design: The title of this section in the outline is *data security measures*, where the technical, physical and organizational measures in place to protect personal data have to be specified. The GDPR in Article 25 (*Privacy by Design*) taking into account a set of issues—such as the costs of implementation, as well as the risks of varying likelihood and severity for rights and freedoms of natural persons—prescribes implementing appropriate technical and organizational measures to meet the requirements (stipulated in the regulation) and protect the rights of data subjects. Therefore, we propose to change the title of this section into *ensuring Privacy by Design*, and include three distinct subsections to specify:

- **4.1-Design violations and threats:** One preliminary step to identify appropriate controls is to identify the design violations and the threats. In this thesis, we introduced a model-based methodology to identify the violations and threats (Section 5.3.3). In a PLA, the threats and the violations have to be clearly documented.

- **4.2-Privacy targets and requirements at risk:** The current PLA outline includes a set of requirements that have to be ensured. However, the currently listed requirements do not fully support the principles relating to the processing of personal data (Article 5 of the GDPR). For instance, *fairness*, *data minimization*, and *storage limitation* are not included. Therefore, we propose to include the list of privacy targets (Table 5.2) in addition to the listed requirements.

  Furthermore, we require to explicitly document the requirements and the privacy targets that are in danger (in regard to the design violations, and threats). See the impact assessment methodology introduced in Section 5.3.4.

- **4.3-Controls and measures:** Eventually, the list of appropriate controls (Section 5.3.5), and their realization by privacy features (Chapter 6), to protect personal data and mitigate the risks have to be documented.

Appendix D

Solove's Harmful Activities, the Privacy Targets, and the Analysis Means

In [158], the authors evaluated the privacy targets concerning their impact on harmful activities introduced by Solove [186]. This evaluation relies on verifying whether privacy harmful activity is likely to occur if the privacy targets are properly addressed. The final result of their work is a table showing how privacy targets tackle activities that can create harm. We extend their table by adding our three new privacy targets, showing in Figure D.1.

Furthermore, in Chapter 5, we provided two tables: 5.1 and 5.2. The former provides a mapping between the Solove's harmful activities and the privacy/security checks. The Latter introduces a mapping between privacy targets and the analysis means (mainly privacy/security checks), assessing the impacts of analysis results on privacy targets—enables identifying the privacy targets *at risk*. In Figure D.1, we further extend the existing result of Oetzel et al. [158], by adding the analysis means to their table.

Analysis means	Decisional Interference	Intrusion	Distortion	Appropriation	Blackmail	Increased Accessibility	Exposure	Disclosure	Breach of Confidentiality	Exclusion	Secondary Use	Insecurity	Identification	Aggregation	Interrogation	Surveillance	A Taxonomy of privacy by Solove Privacy Targets
Existence of PLA	×	×		×			×			×					×	×	P1.1
Purpose check	×	×					×								×	×	P1.2
<<objective>>						×		×			×	×					P1.3
Purpose check						×		×			×	×					P1.4
<<dataPrivacy>>												×	×	×	×	×	P1.5
Purpose check													×	×			P1.6
UMLsec checks			×														P1.7
Retention check												×	×				P1.8
category tag														×			P1.9
Existence of PLA	×					×				×		×	×	×			P1.10
Privacy checks														×			P2.1
Privacy checks								×						×			P2.2
Existence of PLA															×	×	P3.1
Existence of PLA															×		P3.2
Privacy check										×							P4.1
Retention check			×							×							P4.2
Visibility check			×							×							P4.3
Retention check			×														P4.4
Privacy check										×	×						P5.1
Purpose check (Specifically for marketing purpose)		×								×	×						P5.2
Visibility check						×		×									P5.3
Existence of PLA	×																P5.4
Existence of PLA	×		×														P5.5
Security checks			×	×	×	×		×	×			×	×			×	P6.1
Purpose check (notification purpose)							×	×	×			×					P6.2
Existence of PLA			×	×				×	×			×				×	P6.3
Existence of PLA									×			×					P7.1

Figure D.1: Solove's harmful activities, the privacy targets, and the analysis means.

Appendix E

A Mapping Between NIST Privacy Controls and the Privacy Targets

In Chapter 5, we described our model-based privacy impact assessment (PIA) to identify the privacy risks—the privacy targets at risk. Furthermore in Chapter 6, we proposed a system model privacy enhancement based on the NIST privacy controls. The enhancement requires to perform a PIA upfront.

One important step in a PIA (or generally a risk analysis) is to suggest a set of appropriate controls to mitigate the identified risks. As mentioned in 5.3.5, we provide a catalog of privacy and security controls. In this catalog, we mapped the privacy targets and the security requirements to the controls. After performing a PIA or a risk analysis, and identifying the risks, this mapping enables the identification of appropriate controls to mitigate the arising risks.

In Table E.1, we show an excerpt of this catalog focusing only on the mapping between the NIST privacy controls and the privacy targets.

Table E.1: A mapping between privacy targets and the NIST privacy controls

Privacy target	**NIST Controls**
P1.1 Ensuring fair and lawful processing by transparency	TR-1, TR-2, TR-3
P1.2 Ensuring processing only for legitimate purposes	AR-1, AR-2, AR-7, IP-1, UL-1
P1.3 Providing purpose specification	AP-2
P1.4 Ensuring limited processing for specified purposes	AP-2, DM-1
P1.5 Ensuring data avoidance	DM-1, DM-2
P1.6 Ensuring data minimization	DM-1, DM-2, DM-3
P1.7 Ensuring data quality, accuracy and integrity	DI-1, DI-2
P1.8 Ensuring limited storage	DM-2, AP-1, IP-1
P1.9 Ensuring the categorization of personal data	AR-2, AR-3, AR-7
P1.10 Ensuring the prevention of discriminatory effects on natural persons (fairness)	AP-2, DM-1, SE-1
P2.1 Ensuring legitimacy of personal data processing	AR-1, AR-5, AR-7
P2.2 Ensuring legitimacy of sensitive personal data processing	AR-1, AR-5, AP-7
P3.1 Adequate information in case of direct collection of data	AP-2, AR-6, AR-8, TR-1, TR-2, TR-3
P3.2 Adequate information where data is not obtained directly	AP-2, AR-6, AR-8, TR-1, TR-2, TR-3
P4.1 Facilitating the provision of information about processed data and purpose	SE-1, TR-1, TR-3
P4.2 Facilitating the rectification, erasure or blocking of data	DM-2
P4.3 Facilitating the portability of data	IP-2, IP-3
P4.4 Facilitating the notification to third parties about rectification, erasure and blocking of data	UL-2
P5.1 Facilitating the objection to the processing of data	AP-1, IP-2, IP-3
P5.2 Facilitating the objection to direct marketing activities	DM-1, TR-1
P5.3 Facilitating the objection to data-disclosure to others	IP-(1-4)
P5.4 Facilitating the objection to decisions on automated processing	IP-(1-4)
P5.5 Facilitating the data subjects right to dispute the correctness of machine conclusions	IP-(1-4)
P6.1 Ensuring the confidentiality, integrity, availability, and resilience	SE-1, SE-2, AR-1, DI-1, DI-2
P6.2 Ensuring the detection of personal data breaches and their communication to data subjects	TR-1, AR-8, AR-6, AR-2
P6.3 Ensuring the effectiveness of technical and organizational measures. A 32.1(d)	TR-3, IP-4, DI-2
P7.1 Ensuring the accountability	AR-3, AR-8

Appendix F

Privacy Features Including Design (Sub-) Strategies, Patterns, and PETs

In Chapter 6, we proposed to use a feature model to capture the extensive variety of privacy controls, design strategies, patterns, and privacy enhancing technologies (PET), and their interrelations. This feature model is based on a mapping between the NIST controls, a set of privacy (sub-) strategies[1], privacy patterns and PETs.

The mapping between the NIST controls and the privacy (sub-) strategies is demonstrated in Table 6.1. In Section 6.4.1, we described that, we benefit from the correlations of the strategies and patterns, introduced in [52, 53], to map the design strategies to privacy design patterns. We refine this correlation using a number of design patterns obtained from [21, 80, 113, 170, 183, 185]. Finally we map the design patterns to several relevant PET(s) [36, 58, 73, 83, 201]. In Tables F.1-F.8, these mappings are presented.

[1] We reuse a selection of eight privacy design strategies [108], including their concrete specifications using *sub-strategies* [52]

Table F.1: The privacy features of the *Minimize* strategy

Sub Strategy	**Pattern**	**PET**
Exclude	Protection against Tracking	Single proxies, VPNs, restrict cookies, disable cookies, Avoid use of third-party libraries, Do not store identifiers of smart cards, Reduce time and location granularity on device, Cluster IoT data streams and only release clusters with at least k members, Avoid storing encryption keys on device, Abine, Ad blockers, Anonymouse, Internet proxy, Ixquick, MyTube
Select	Partial identification	Extract relevant features on sensor, Cluster IoT data streams and only release clusters with at least k members
Strip	Strip invisible metadata, Authentication, Attribute-based credentials	Statistical disclosure control, Avoid storing encryption keys on device, Authenticate users without identifying them, Use short-lived pseudonyms for car-to-car communication, IRMA
Destroy	Limited data retention	discard raw data, Eliminate mapping between short-term and long-term identifiers, Ccleaner, Eraser

Table F.2: The privacy features of the *Hide* strategy

Sub Strategy	Pattern	PET
Restrict	Aggregation gateway, Anonymous reputation-based Blacklisting, Attribute-based credentials, Selective access control, Trustworthy privacy plug-in	Single proxies VPNs, Use cryptographically enforced role-based access control, Use identity-based encryption for private service discovery, Use attribute-based encryption for access control (RABAC), Authenticate users based on attributes instead of identities
Mix	Anonymity Set, Onion routing	Single proxies, VPNs, Tor service, Mixmaster, Mixminion, Broadcast, Steganography, Watermarking, Anonymous remailer (awxcnx), Randomize browser fingerprints, Ensure k-anonymity of sensor readings, Ensure spatio-temporal readings cover at least k individuals, GoogleSharing, I2P, JonDo
Obfuscate	Added-noise measurement obfuscation, Anonymity set, Encryption-user-managed-keys, Use of dummies,	Single Proxies, VPNs, (TrueCrypt)Steganography, Differential privacy, Verifiable encryption, Lying, Release noisy aggregates of data, Obfuscate locations with planar Laplace noise, Apply noise to meter readings, Ensure correct usage of SSL/TLS with static analysis, Ensure correct usage of SSL/TLS with dynamically linked libraries, Secure public WiFi with WPA2, RetroShare
Dissociate	Pseudonymous identity, Pseudonymous messaging	Onion routing, RFID deactivation, Safe Harbor, Anonymouse, Hosts file domain blocking, Java anon proxy

Table F.3: The privacy features of the *Separate* strategy

Sub Strategy	**Pattern**	**PET**
Distribute	Private link	Store private data only on private devices, Use secure distributed data storage
Isolate	Personal data store, User data confinement pattern, Physical privacy zones	Implement Sensors only in specific places and inform data subjects, Smart meter, Pay as you drive, Isolate sensors from other systems, Separate entities that ask for and receive sensor readings, Process data for smart metering on device, Process data for toll pricing on device, Hide access patterns to remote files, Hide access patterns to remote databases, Hosts file domain blocking, Privatix live-system

Table F.4: The privacy features of the *Abstract* strategy

Sub Strategy	**Pattern**	**PET**
Summarize	Statistical disclosure control, Aggregation, Anonymity set	K anonymity, Differential privacy, l-diversity, Query restriction, Sampling (Non-perturbative masking), Microaggregation, Release only data that satisfy k-anonymity, Ensure k-anonymity of sensor readings, Ensure spatio-temporal readings over at least k individuals, Use privacy-preserving data aggregation Aggregate data over multiple participants privately
Group	Location granularity, Aggregation, Generalization	Top/bottom coding, Local suppression, Microaggregation, Reduce time and location granularity on device, Aggregate sensor readings from multiple participants privately, Aggregate data over multiple participants, e.g. energy consumption, Use privacy-preserving data aggregation,

Table F.5: The privacy features of the *Inform* strategy

Sub Strategy	**Pattern**	**PET**
Supply	Policy matching display, Privacy-aware network client, Privacy dashboard, Privacy icons	Privacy dashboard, RFID Logo, Privacy level agreements, Platform for privacy preferences (P3P) , Privacy bird, Formal Framework: CI (Contextual Integrity), S4P, SIMPL, Pidder
Notify	Ambient notice, Asynchronous notice, Data breach notification pattern, Handling unusual account activities with multiple factors, Privacy mirrors, Who is listening	Lightbeam (Firefox), TaintDroid, Mobilitics, Data track, Certificate patrol, Electronic mail
Explain	Layered policy design, Privacy-aware network client, Privacy color coding, Privacy icons, On-demand explanation	ToSDR, TOSBack, RFID Logo, Privacy level agreements (VisiOn Project), Privacy bird, PRIME, Panopticlick

Table F.6: The privacy features of the *Control* strategy

Sub Strategy	Pattern	PET
Consent	Incentivized participation, Informed consent for web-based transaction, Lawful consent, Obtaining explicit consent, Outsourcing [with consent], Sign an agreement to Solve lack of trust	Establishing PLA (organizational level), JITCTAs (Just-In-Time-Click-Through Agreements), Drag-and.Drop agreements (DADAs), Separate purposes (PLAs), Personalized negotiation, PrivacyFinder
Choose	Buddy list, Discouraging blanket strategies, Incentivized participation, Negotiation of privacy policy, Pay Back, Private link, Selective access control, Single point of contact, Selective disclosure, Blurred personal data, Attention screen, Reciprocity	Compute statistics over sensor readings from participants privately, RABAC (for Buddy List), Selective access control, SPoC, c-Consent, TUKAN, Anonymous access, BugMeNot, TeamSpace, WebWasher
Update	Active broadcast of presence, Enable/Disable functions, Reasonable level of control	Change device identifiers frequently to prevent fingerprinting, Data track, check-in model (broadcasting status), PLA (VisiOn Privacy Platform)
Retract	Decoupling content and location information visibility, Masquerade	(Exclude) Reduce time and location granularity on device, (Exclude) cluster IoT data streams and only release clusters with at least k members, Implement levels of publicity, Anonymous access, BleachBit, Ccleaner

Table F.7: The privacy features of the *Enforce* strategy

Sub Strategy	Pattern	PET
Create	Federated privacy impact assessment, Fair information practices, Privacy-sensitive architectures	Perform model-based PIA, Freenet, HTTPs everywhere
Maintain	Appropriate privacy feedback	PLA, KeePass
Uphold	Identity federation do not track pattern, Obligation management, Distributed usage control	Enforce honesty of device for local processing, Enforce honesty of vehicle for local processing, Implement an orchestrator (a Javascript program as a App in client browser, the orchestrator makes sure that the identity broker can't correlate the original request from the service provider with the assertions that are returned from the identity provider.), Sticky policies, EPAL

Table F.8: The privacy features of the *Demonstrate* strategy

Sub Strategy	Pattern	PET
Audit	Audit interceptor	Java API for logging Log4J, Ghostery
Log	Secure logger, Data access log, Birds of feather	Secure data logger strategy, Secure log store strategy, Stackdriver logging, MEMOIR, Autonomy CEN, Yenta
Report	Dissemination of privacy Program information	ClouDAT platform, VisiOn privacy platform, Ghostery

Appendix G

A Walk Through of CARiSMA's Screencast

We first describe how CARiSMA may be installed, afterwards, we provide a walk through of the screencast. All required information on CARiSMA may be found on the CARiSMA website:

- **CARiSMA website**: `http://carisma.umlsec.de`
- **Screencast**: `https://youtu.be/b5zeHig3ARw`
- **Evaluation data (Municipality of Athens' case study)**: `http://carisma.umlsec.de/conferences/FSE2017/CarismaFSE2017EvaluationData.zip`

G.1 Tool Installation

CARiSMA may be installed on the latest version of Eclipse (*Help* → *Install New Software...*) from the CARiSMA update-site (`http://carisma.umlsec.de/updatesite`). Please select the *Vision* feature from the *CARiSMA Project Specific Features* category. We suggest to download Eclipse Modeling Tools, which includes all necessary plugins for CARiSMA. Moreover, we recommend to install and use the Papyrus editor to model the UML diagrams.

G.2 Help Content

For CARiSMA a user manual is provided which can be found in the *help content* of Eclipse. After installing CARiSMA, the manual is available under: *Help → Help Contents → CARiSMA*.

G.3 The Script of the Screencast

In the screencast we mainly show the new functionalities that are added to CARiSMA in the course of VisiOn EU project. Therefore, in the first step, we briefly introduce the VisiOn Privacy Platform and it's main components.

In the CARiSMA demonstration we mainly perform the following steps. In brackets we have indicated the respective timestamps in the video.

- **Perform a pre-analysis for a system model (2:15)**: A system model either is modeled by a system designer, or already exists in the system specifications, and is imported to Eclipse as an existing project. For this system model, a CARiSMA analysis is manually created by pressing the right-hand button of a mouse on the model and selecting *New → CARiSMA → Analysis*. To perform a pre-analysis, in the created CARiSMA analysis, the *Create Help Document for STS mapping* action must be first selected from the list of available actions (checks), and then be run.

- **Generate automatically a CARiSMA analysis (2:42)**: The result of the pre-analysis is provided in the *Analysis Results* view. From the result of a pre-analysis, a CARiSMA analysis may be automatically generated by pressing the right-hand button of a mouse and then selecting *Create automated analysis from help document*. The generated CARiSMA analysis contains the proper actions that must be performed on the system model. However, it is possible to add more checks to the generated CARiSMA analysis.

- **Generate a help report (3:12)**: In addition to the previous step, in the analysis results by pressing the right-hand button of a mouse and then selecting *Create report for selected analysis*, a help report will be generated. This report assists a system designer to annotate the system models with the security and privacy profiles. It contains **(I)** a mapping between the elements of a STS model and the elements of a selected system model, **(II)** The UML diagrams that must be annotated, and **(III)** the relevant security and privacy profiles that must be applied to the UML diagrams.

- **Annotate the system model (3:51)**: The security and privacy profiles are used to annotate the models. The annotations enable the security and privacy checks of CARiSMA.

- **Perform an analysis (5:04)**: After completing the annotation of the system model, the generated CARiSMA analysis may be performed on the system model. The results are presented in the *Analysis Results* view.

Appendix H

Lebenslauf

Persönliche Daten

Name	Amirshayan Ahmadian
Geburtsdatum	23.03.1986
Geburtsort	Tehran, Iran
Staatsangehörigkeit	deutsch
Kontakt	ahmadian@uni-koblenz.de +49 261 287 2768

Schulbildung

1992 – 2000	Grundschule Mohit in Teheran
2000 – 2004	Gymnasium Sadr in Tehran Abschluss: Abitur

Hochschulausbildung

2004 – 2009	Studium an der Universität Science & Culture Teheran Studiengang: Software Engineering Abschluss: Bachelor of Science
2010 – 2012	Studium an der Universität Paderborn Studiengang: Informatik Vertiefungsgebiet: Softwaretechnik und Informationssysteme Masterthesis: Exploiting Planning Graphs and Landmarks for Efficient GTS Planning Abschluss: Master of Science

Berufserfahrung

06/2011 – 03/2012	Studentische Hilfskraft, Universität Paderborn Arbeitsgruppe „Spezifikation und Modellierung von Softwaresystemen“ (Prof. Dr. Heike Wehrheim)
04/2013 – 04/2014	Spezialist für Softwareentwicklung, ARAG Versicherung München
04/2014 – 10/2015	Wissenschaftlicher Mitarbeiter, TU Dortmund Arbeitsgruppe „Software Engineering“ (Prof. Dr. Jan Jürjens)
10/2015 – Heute	Wissenschaftlicher Mitarbeiter, Universität Koblenz-Landau Arbeitsgruppe „Software Engineering“ (Prof. Dr. Jan Jürjens)

Sprachkompetenzen

Deutsch fließend in Wort und Schrift
Englisch fließend in Wort und Schrift
Persisch (Muttersprache)

Bibliography

[1] What is AMKA? `http://www.amka.gr/tieinai_en.html`. Accessed: 2019-06-01.

[2] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. Hippocratic databases. In *VLDB 2002, Proceedings of 28th International Conference on Very Large Data Bases, August 20-23, 2002, Hong Kong, China*, pages 143–154, 2002.

[3] Amir Shayan Ahmadian and Jan Jürjens. Supporting model-based privacy analysis by exploiting privacy level agreements. In *2016 IEEE International Conference on Cloud Computing Technology and Science, CloudCom 2016, Luxembourg, December 12-15, 2016*, pages 360–365, 2016.

[4] Amir Shayan Ahmadian, Daniel Strüber, Jan Jürjens, and Volker Riediger. Model-based privacy analysis in industrial ecosystems: A formal foundation. *International Journal on Software and Systems Modeling*. Submitted.

[5] Amir Shayan Ahmadian, Fabian Coerschulte, and Jan Jürjens. Supporting the security certification and privacy level agreements in the context of clouds. In *Business Modeling and Software Design - 5th International Symposium, BMSD 2015, Milan, Italy, July 6-8, 2015, Revised Selected Papers*, pages 80–95, 2015.

[6] Amir Shayan Ahmadian, Sven Peldszus, Qusai Ramadan, and Jan Jürjens. Model-based privacy and security analysis with CARiSMA. In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering, ESEC/FSE 2017, Paderborn, Germany, September 4-8, 2017*, pages 989–993, 2017.

[7] Amir Shayan Ahmadian, Daniel Strüber, Volker Riediger, and Jan Jürjens. Model-based privacy analysis in industrial ecosystems. In *Modelling Foundations and Applications - 13th European Conference, ECMFA 2017, Held as Part of STAF 2017, Marburg, Germany, July 19-20, 2017, Proceedings*, pages 215–231, 2017.

[8] Amir Shayan Ahmadian, Jan Jürjens, and Daniel Strüber. Extending model-based privacy analysis for the industrial data space by exploiting privacy

level agreements. In *Proceedings of the 33rd Annual ACM Symposium on Applied Computing, SAC 2018, Pau, France, April 09-13, 2018*, pages 1142–1149, 2018.

[9] Amir Shayan Ahmadian, Daniel Strüber, Volker Riediger, and Jan Jürjens. Supporting privacy impact assessment by model-based privacy analysis. In *Proceedings of the 33rd Annual ACM Symposium on Applied Computing, SAC 2018, Pau, France, April 09-13, 2018*, pages 1467–1474, 2018.

[10] Amir Shayan Ahmadian, Daniel Strüber, and Jan Jürjens. Privacy-enhanced system design modeling based on privacy features. In *Proceedings of the 34th Annual ACM Symposium on Applied Computing, SAC 2019, Limassol, Cyprus, April 08-12, 2019*, pages 1492–1499, 2019.

[11] Aj Albrecht. Measuring application development productivity. In I. B. M. Press, editor, *IBM Application Development Symp.*, pages 83–92, October 1979.

[12] Azadeh Alebrahim, Denis Hatebur, and Ludger Goeke. Pattern-based and ISO 27001 compliant risk analysis for cloud systems. In *IEEE 1st Workshop on Evolving Security and Privacy Requirements Engineering, ESPRE 2014, 25 August, 2014, Karlskrona, Sweden*, pages 42–47, 2014.

[13] Majed Alshammari and Andrew Simpson. A UML profile for privacy-aware data lifecycle models. In *Computer Security - ESORICS 2017 International Workshops, CyberICPS 2017 and SECPRE 2017, Oslo, Norway, September 14-15, 2017, Revised Selected Papers*, pages 189–209, 2017.

[14] Konstantinos Angelopoulos, Vasiliki Diamantopoulou, Haralambos Mouratidis, Michalis Pavlidis, Mattia Salnitri, Paolo Giorgini, and José F. Ruiz. A holistic approach for privacy protection in E-government. In *Proceedings of the 12th International Conference on Availability, Reliability and Security, Reggio Calabria, Italy, August 29 - September 01, 2017*, pages 17:1–17:10, 2017.

[15] Thibaud Antignac and Daniel Le Métayer. Privacy by Design: From technologies to architectures - (Position Paper). In *Privacy Technologies and Policy - Second Annual Privacy Forum, APF 2014, Athens, Greece, May 20-21, 2014. Proceedings*, pages 1–17, 2014.

[16] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy H. Katz, Andrew Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. Above the clouds: A Berkeley view of cloud computing. Technical Report UCB/EECS-2009-28, EECS Department, University of California, Berkeley, Feb 2009.

[17] Paul Ashley, Satoshi Hada, Günter Karjoth, Calvin Powers, and Matthias Schunter. Enterprise privacy authorization language (EPAL). *IBM Research*, 2003.

[18] Sören Auer, Jan Jürjens, Boris Otto, Gerd Brost, Christoph Lange, Christoph Quix, Jan Cirullies, Steffen Lohmann, Jochen Schon, Andreas Eitel, Christian Mader, Daniel Schulz, Thilo Ernst, Nadja Menz, Julian Schütte, Christian Haas, Ralf Nagel, Markus Spiekermann, Manuel Huber, Heinrich Pettenpohl, Sven Wenzel, Christian Jung, and Jaroslav Pullmann. Reference architecture model for the Industrial Data Space. White paper, Fraunhofer, 2018.

[19] Vanessa Ayala-Rivera and Liliana Pasquale. The grace period has ended: An approach to operationalize GDPR requirements. In *26th IEEE International Requirements Engineering Conference, RE 2018, Banff, AB, Canada, August 20-24, 2018*, pages 136–146, 2018.

[20] Monir Azraoui, Kaoutar Elkhiyaoui, Melek Önen, Karin Bernsmed, Anderson Santana de Oliveira, and Jakub Sendor. A-PPL: an accountability policy language. In *Data Privacy Management, Autonomous Spontaneous Security, and Security Assurance - 9th International Workshop, DPM 2014, 7th International Workshop, SETOP 2014, and 3rd International Workshop, QASA 2014, Wroclaw, Poland, September 10-11, 2014. Revised Selected Papers*, pages 319–326, 2014.

[21] Harun Baraki, Kurt Geihs, Axel Hoffmann, Christian Voigtmann, Romy Kniewel, Björn Elmar Macek, and Julika Zirfas. Towards interdisciplinary design patterns for ubiquitous computing applications. Technical report, Kassel, Germany: Kassel University Press GmbH,, 2014.

[22] Ken Barker, Mina Askari, Mishtu Banerjee, Kambiz Ghazinour, Brenan Mackas, Maryam Majedi, Sampson Pun, and Adepele Williams. A data privacy taxonomy. In *Dataspace: The Final Frontier, 26th British National Conference on Databases, BNCOD 26, Birmingham, UK, July 7-9, 2009. Proceedings*, pages 42–54, 2009.

[23] Tânia Basso, Leonardo Montecchi, Regina Moraes, Mario Jino, and Andrea Bondavalli. Towards a UML profile for privacy-aware applications. In *15th IEEE International Conference on Computer and Information Technology, CIT 2015; 14th IEEE International Conference on Ubiquitous Computing and Communications, IUCC 2015; 13th IEEE International Conference on Dependable, Autonomic and Secure Computing, DASC 2015; 13th IEEE International Conference on Pervasive Intelligence and Computing, PICom 2015, Liverpool, United Kingdom, October 26-28, 2015*, pages 371–378, 2015.

[24] Jörg Becker, Patrick Delfmann, Hanns-Alexander Dietrich, Matthias Steinhorst, and Mathias Eggert. Business process compliance checking - applying and evaluating a generic pattern matching approach for conceptual models in the financial sector. *Information Systems Frontiers*, 18(2):359–405, 2016.

[25] Kristian Beckers, Holger Schmidt, Jan-Christoph Küster, and Stephan Faßbender. Pattern-based support for context establishment and asset identification of the ISO 27000 in the field of cloud computing. In *Sixth International Conference on Availability, Reliability and Security, ARES 2011, Vienna, Austria, August 22-26, 2011*, pages 327–333, 2011.

[26] Kristian Beckers, Stephan Faßbender, Maritta Heisel, and Rene Meis. A problem-based approach for computer-aided privacy threat identification. In *Privacy Technologies and Policy - First Annual Privacy Forum, APF 2012, Limassol, Cyprus, October 10-11, 2012, Revised Selected Papers*, pages 1–16, 2012.

[27] Kristian Beckers, Maritta Heisel, Isabelle Côté, Ludger Goeke, and Selim Güler. Structured pattern-based security requirements elicitation for clouds. In *2013 International Conference on Availability, Reliability and Security, ARES 2013, Regensburg, Germany, September 2-6, 2013*, pages 465–474, 2013.

[28] Kristian Beckers, Isabelle Côté, and Ludger Goeke. A catalog of security requirements patterns for the domain of cloud computing systems. In *Symposium on Applied Computing, SAC 2014, Gyeongju, Republic of Korea - March 24 - 28, 2014*, pages 337–342, 2014.

[29] Victoria Bellotti and Abigail Sellen. Design for privacy in ubiquitous computing environments. In *Third European Conference on Computer Supported Cooperative Work, ECSCW'93, Milano, September 13-17, 1993, Proceedings*, page 75, 1993.

[30] Salima Benbernou, Hassina Meziane, Yin Hua Li, and Mohand-Said Hacid. A privacy agreement model for web services. In *2007 IEEE International Conference on Services Computing (SCC 2007), 9-13 July 2007, Salt Lake City, Utah, USA*, pages 196–203, 2007.

[31] Colin J. Bennett. *Regulating Privacy: Data Protection and Public Policy in Europe and the United States*. Cornell paperbacks. Cornell University Press, 1992. ISBN 9780801480102.

[32] Alastair R. Beresford and Frank Stajano. Mix zones: User privacy in location-aware services. In *2nd IEEE Conference on Pervasive Computing and Communications Workshops (PerCom 2004 Workshops), 14-17 March 2004, Orlando, FL, USA*, pages 127–131, 2004.

[33] Vieri Del Bianco, Luigi Lavazza, and Sandro Morasca. A proposal for simplified model-based cost estimation models. In *Product-Focused Software Process Improvement - 13th International Conference, PROFES 2012, Madrid, Spain, June 13-15, 2012 Proceedings*, pages 59–73, 2012.

[34] Felix Bieker, Michael Friedewald, Marit Hansen, Hannah Obersteller, and Martin Rost. A process for data protection impact assessment under the European General Data Protection Regulation. In *Privacy Technologies and Policy - 4th Annual Privacy Forum, APF 2016, Frankfurt/Main, Germany, September 7-8, 2016, Proceedings*, pages 21–37, 2016.

[35] Christoph Bier, Pascal Birnstill, Erik Krempel, Hauke Vagts, and Jürgen Beyerer. Enhancing privacy by design from a developer's perspective. In *Privacy Technologies and Policy - First Annual Privacy Forum, APF 2012, Limassol, Cyprus, October 10-11, 2012, Revised Selected Papers*, pages 73–85, 2012.

[36] John J. Borking and Charles D. Raab. Laws, PETs and other technologies for privacy protection. *Journal of Information, Law and Technology*, 2001(1), 2001.

[37] Ruth Breu, Klaus Burger, Michael Hafner, Jan Jürjens, Gerhard Popp, Guido Wimmel, and Volkmar Lotz. Key issues of a formally based process model for security engineering. In *Sixteenth International Conference "Software & Systems Engineering & their Applications"*, Paris, 2003.

[38] Bundesamt für Sicherheit in der Informationstechnik. IT-Grundschutz Methodology, BSI Standard 100-2, 2008.

[39] Bundesamt für Sicherheit in der Informationstechnik. BSI-Grundschutz Katalog, 2016.

[40] Shawn A. Butler. Security attribute evaluation method: A cost-benefit approach. In *Proceedings of the 24th International Conference on Software Engineering, ICSE 2002, 19-25 May 2002, Orlando, Florida, USA*, pages 232–240, 2002.

[41] Ji-Won Byun, Elisa Bertino, and Ninghui Li. Purpose based access control of complex data for privacy protection. In *10th ACM Symposium on Access Control Models and Technologies, SACMAT 2005, Stockholm, Sweden, June 1-3, 2005, Proceedings*, pages 102–110, 2005.

[42] Ann Cavoukian and Michelle Chibba. Advancing privacy and security in computing, networking and systems innovations through privacy by design. In *Proceedings of the 2009 conference of the Centre for Advanced Studies on Collaborative Research, November 2-5, 2009, Toronto, Ontario, Canada*, pages 358–360, 2009.

[43] Ilia Christantoni, Claudio Biffi, Dimitri Bonutto, and Andres Castillo Sanz. VisiOn Pilots Reports - VisiOn Project Deliverable 5.2. Technical report, 2017.

[44] Ilia Christantoni, Andrea Praitano, Amir Shayan Ahmadian, Mauro Brunato, Julian Flake, Mattia Salnitri, Konstantinos Angelopoulos, and Vasiliki Diamantopoulou. Training Activities Manual - VisiOn Project Deliverable 6.3. Technical report, 2017.

[45] Roger Clarke. An evaluation of privacy impact assessment guidance documents. *International Data Privacy Law*, 1(2):111, 2011.

[46] Cloud Security Alliance. Security guidance for critical areas of focus in cloud computing V3.0. `https://downloads.cloudsecurityalliance.org/initiatives/guidance/csaguide.v3.0.pdf`. Accessed: 2019-06-01.

[47] Cloud Security Alliance. The notorious nine cloud computing top threats in 2013. `https://cloudsecurityalliance.org/download/the-notorious-nine-cloud-computing-top-threats-in-2013/`, February 2013. Accessed: 2019-06-01.

[48] Cloud Security Alliance. Privacy Level Agreement [V1]: PLA outline for the sale of cloud services in the European Union, 2013.

[49] Cloud Security Alliance. Privacy Level Agreement [V2]: A compliance tool for providing cloud services in the European Union, 2015.

[50] Cloud Security Alliance. Top treacherous twelve' cloud computing top threats in 2016. `https://cloudsecurityalliance.org/download/the-treacherous-twelve-cloud-computing-top-threats-in-2016/`, February 2016. Accessed: 2019-06-01.

[51] Cloud Security Alliance. Cloud control matrix. `https://cloudsecurityalliance.org/artifacts/csa-ccm-v-3-0-1-11-12-2018-FINAL/`, 2018. Accessed: 2019-06-01.

[52] Michael Colesky, Jaap-Henk Hoepman, and Christiaan Hillen. A critical analysis of privacy design strategies. In *2016 IEEE Security and Privacy Workshops, SP Workshops 2016, San Jose, CA, USA, May 22-26, 2016*, pages 33–40, 2016.

[53] Michael Colesky, Julio C. Caiza, José M. del Álamo, Jaap-Henk Hoepman, and Yod-Samuel Martín. A system of privacy patterns for user control. In *Proceedings of ACM SAC Conference (SAC18)*, New York, NY, USA, 2018. ACM.

[54] Pietro Colombo and Elena Ferrari. Towards a modeling and analysis framework for privacy-aware systems. In *2012 International Conference on Privacy, Security, Risk and Trust, PASSAT 2012, and 2012 International Confernece on Social Computing, SocialCom 2012, Amsterdam, Netherlands, September 3-5, 2012*, pages 81–90, 2012.

[55] Pietro Colombo and Elena Ferrari. Enforcement of purpose based access control within relational database management systems. *IEEE Trans. Knowl. Data Eng.*, 26(11):2703–2716, 2014.

[56] Isabelle Côté, Maritta Heisel, Holger Schmidt, and Denis Hatebur. UML4PF - A tool for problem-oriented requirements analysis. In *RE 2011, 19th IEEE International Requirements Engineering Conference, Trento, Italy, August 29 2011 - September 2, 2011*, pages 349–350, 2011.

[57] Gordana Dodig Crnkovic. *Constructive Research and Info-computational Knowledge Generation*, pages 359–380. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. ISBN 978-3-642-15223-8.

[58] George Danezis, Josep Domingo-Ferrer, Marit Hansen, Jaap-Henk Hoepman, Daniel Le Métayer, Rodica Tirtea, and Stefan Schiffner. Privacy and data protection by design - from policy to engineering. *CoRR*, abs/1501.03726, 2015.

[59] Data Protection Act. Conducting privacy impact assessments code of practice. `https://iapp.org/media/pdf/resource_center/ICO_pia-code-of-practice.pdf`, 2014. Accessed: 2019-06-01.

[60] Data Protection Authorities of Greece and Germany, Clara Galan Manso, and Slawomir Gorniak. Recommendations for a methodology of the assessment of severity of personal data breaches. Technical report, The European Union Agency for Network and Information Security, 2013.

[61] Sourya Joyee De and Daniel Le Métayer. PRIAM: A privacy risk analysis methodology. In *Data Privacy Management and Security Assurance - 11th International Workshop, DPM 2016 and 5th International Workshop, QASA 2016, Heraklion, Crete, Greece, September 26-27, 2016, Proceedings*, pages 221–229, 2016.

[62] Sourya Joyee De and Daniel Le Métayer. Privacy harm analysis: A case study on smart grids. In *2016 IEEE Security and Privacy Workshops, SP Workshops 2016, San Jose, CA, USA, May 22-26, 2016*, pages 58–65, 2016.

[63] Patrick Delfmann, Sebastian Herwig, Lukasz Lis, Armin Stein, Katrin Tent, and Jörg Becker. Pattern specification and matching in conceptual models - A generic approach based on set operations. *Enterprise Modelling and Information Systems Architectures*, 5(3):24–43, 2010.

[64] Folker den Braber, Ida Hogganvik, Mass Soldal Lund, Ketil Stølen, and Fredrik Vraalsen. Model-based security analysis in seven steps — a guided tour to the CORAS method. *BT Technology Journal*, 25(1):101–117, 2007. ISSN 1573-1995.

[65] Mina Deng, Kim Wuyts, Riccardo Scandariato, Bart Preneel, and Wouter Joosen. A privacy threat analysis framework: Supporting the elicitation and fulfillment of privacy requirements. *Requir. Eng.*, 16(1):3–32, 2011.

[66] Michela D'Errico and Siani Pearson. Towards a formalised representation for the technical enforcement of privacy level agreements. In *2015 IEEE International Conference on Cloud Engineering, IC2E 2015, Tempe, AZ, USA, March 9-13, 2015*, pages 422–427, 2015.

[67] Rinku Dewri, Nayot Poolsappasit, Indrajit Ray, and Darrell Whitley. Optimal security hardening using multi-objective optimization on attack tree models of networks. In *Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA, October 28-31, 2007*, pages 204–213, 2007.

[68] Vasiliki Diamantopoulou, Konstantinos Angelopoulos, Julian Flake, Andrea Praitano, José F. Ruiz, Jan Jürjens, Michalis Pavlidis, Dimitri Bonutto, Andrès Castillo Sanz, Haralambos Mouratidis, Javier Garcia-Robles, and Alberto Eugenio Tozzi. Privacy data management and awareness for public administrations: A case study from the healthcare domain. In *Privacy Technologies and Policy - 5th Annual Privacy Forum, APF 2017, Vienna, Austria, June 7-8, 2017, Revised Selected Papers*, pages 192–209, 2017.

[69] Vasiliki Diamantopoulou, Konstantinos Angelopoulos, Michalis Pavlidis, and Haralambos Mouratidis. A metamodel for GDPR-based privacy level agreements. In *Proceedings of the ER Forum 2017 and the ER 2017 Demo Track co-located with the 36th International Conference on Conceptual Modelling (ER 2017), Valencia, Spain, - November 6-9, 2017.*, pages 285–291, 2017. URL `http://ceur-ws.org/Vol-1979/paper-08.pdf`.

[70] Vasiliki Diamantopoulou, Michalis Pavlidis, and Haralambos Mouratidis. Privacy level agreements for public administration information systems. In *Proceedings of the Forum and Doctoral Consortium Papers Presented at the 29th International Conference on Advanced Information Systems Engineering, CAiSE 2017, Essen, Germany, June 12-16, 2017*, pages 97–104, 2017.

[71] Andreas Drakos, Basilis Barekas, A. Daskalopoulos, Amir Shayan Ahmadian, Jan Jürjens, Qusai Ramadan, Tahir Emre Kalaycı, Mauro Brunato, Roberto Battiti, and Paolo Giorgini. Privacy Specification Component - ViSiOn Project Deliverable 3.4. Technical report, 2016.

[72] François Dupressoir, Andrew D. Gordon, Jan Jürjens, and David A. Naumann. Guiding a general-purpose C verifier to prove cryptographic protocols. *Journal of Computer Security*, 22(5):823–866, 2014.

[73] David Eckhoff and Isabel Wagner. Privacy in the Smart City - applications, technologies, challenges, and solutions. *IEEE Communications Surveys and Tutorials*, 20(1):489–516, 2018.

[74] European Commission. Special Eurobarometer 431 - Data Protection. Technical report, 2015.

[75] European Commission. Question and Answers - General Data Protection Regulation. *European Commission - Press release*, 2018.

[76] European Data Protection Supervisor (EDPS). Preliminary Opinion on Privacy by Design. Technical report, May 2018.

[77] European Network and Information Security Agency. Cloud computing - benefits, risks and recommendations for information security. `https://resilience.enisa.europa.eu/cloud-security-and-resilience/publications/`, 2009. Accessed: 2019-06-01.

[78] European Union. *Charter of Fundamental Rights of the European Union*. European Union, Brussels, 2010.

[79] European Union Agency for Network and Information Security (ENISA). ENISA Threat Landscape Report 2018. Technical report, 2018.

[80] Eduardo Fernandez-Buglioni. *Security Patterns in Practice: Designing Secure Architectures Using Software Patterns*. Wiley Publishing, 1st edition, 2013. ISBN 1119998948, 9781119998945.

[81] Eduardo Fernández-Medina, Jan Jürjens, Juan Trujillo, and Sushil Jajodia. Model-driven development for secure information systems. *Information & Software Technology*, 51(5):809–814, 2009.

[82] Simone Fischer-Hübner. *IT-Security and Privacy - Design and Use of Privacy-Enhancing Security Mechanisms*, volume 1958 of *Lecture Notes in Computer Science*. Springer, 2001.

[83] The Center for Internet and Society. CIS PET wiki. `https://cyberlaw.stanford.edu/wiki/index.php/PET`. Accessed: 2019-06-01.

[84] Robert B. France and Bernhard Rumpe, editors. *«UML»'99: The Unified Modeling Language - Beyond the Standard, Second International Conference, Fort Collins, CO, USA, October 28-30, 1999, Proceedings*, volume 1723 of *Lecture Notes in Computer Science*, 1999. Springer.

[85] Robert B. France and Bernhard Rumpe. Model-driven development of complex software: A research roadmap. In *International Conference on Software Engineering, ISCE 2007, Workshop on the Future of Software Engineering, FOSE 2007, May 23-25, 2007, Minneapolis, MN, USA*, pages 37–54, 2007.

[86] French Data Protection Authority (CNIL). Privacy Impact Assessment (PIA) Methodology. `https://www.cnil.fr/sites/default/files/atoms/files/cnil-pia-1-en-methodology.pdf`, 2018. Accessed: 2019-06-01.

[87] French Data Protection Authority (CNIL). Privacy Impact Assessment (PIA) Knowledge Bases. `https://www.cnil.fr/sites/default/files/atoms/files/cnil-pia-3-en-knowledgebases.pdf`, 2018. Accessed: 2019-06-01.

[88] Rafa Galvez and Seda Gurses. The odyssey: Modeling privacy threats in a brave new world. In *2018 IEEE European Symposium on Security and Privacy Workshops, EuroS&P Workshops 2018, London, United Kingdom, April 23-27, 2018*, pages 87–94, 2018.

[89] Geri Georg, Indrakshi Ray, Kyriakos Anastasakis, Behzad Bordbar, Manachai Toahchoodee, and Siv Hilde Houmb. An aspect-oriented methodology for designing secure applications. *Information & Software Technology*, 51(5):846–864, 2009.

[90] Kambiz Ghazinour, Maryam Majedi, and Ken Barker. A lattice-based privacy aware access control model. In *Proceedings of the 12th IEEE International Conference on Computational Science and Engineering, CSE 2009, Vancouver, BC, Canada, August 29-31, 2009*, pages 154–159, 2009.

[91] Dieter Gollmann. *Computer Security*. John Wiley & Sons, Inc., New York, NY, USA, 1999. ISBN 0-471-97844-2.

[92] Maximilian Gottwald. Sicherheitsanforderungs-Katalog für den Industrial Data Space. Master's thesis, University of Koblenz Landau, January 2019.

[93] Rüdiger Grimm, Daniela Simic-Draws, Katharina Bräunlich, Andreas Kasten, and Anastasia Meletiadou. Referenzmodell für ein Vorgehen bei der IT-Sicherheitsanalyse. *Informatik Spektrum*, 39(1):2–20, 2016.

[94] Nicola Guarino. Formal ontology in information systems: Proceedings of the 1st international conference june 6-8, 1998, trento, italy. In *Proceedings of the 1st International Conference June 6-8, 1998, Trento, Italy*, Amsterdam, The Netherlands, The Netherlands, 1998. IOS Press.

[95] Michele Guerriero, Damian Andrew Tamburri, Youssef Ridene, Francesco Marconi, Marcello M. Bersani, and Matej Artac. Towards DevOps for privacy-by-design in data-intensive applications: A research roadmap. In *Companion Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering, ICPE 2017, L'Aquila, Italy, April 22-26, 2017*, pages 139–144, 2017.

[96] Seda Gürses, Carmela Troncoso, and Claudia Diaz. Engineering privacy by design. 2011.

[97] Seda F. Gürses. Can you engineer privacy? *Commun. ACM*, 57(8):20–23, 2014.

[98] Irit Hadar, Tomer Hasson, Oshrat Ayalon, Eran Toch, Michael Birnhack, Sofia Sherman, and Arod Balissa. Privacy by designers: software developers' privacy mindset. *Empirical Software Engineering*, 23(1):259–289, 2018.

[99] Munawar Hafiz. A pattern language for developing privacy enhancing technologies. *Softw., Pract. Exper.*, 43(7):769–787, 2013.

[100] Jay M. Handelman and Stephen J. Arnold. The role of marketing actions with a social dimension: Appeals to the institutional environment. *Journal of Marketing*, 63(3):33–48, 1999. ISSN 00222429.

[101] Denis Hatebur. *Pattern- and Component-based Development of Dependable Systems*. PhD thesis, 2012.

[102] Michael Haus, Muhammad Waqas, Aaron Yi Ding, Yong Li, Sasu Tarkoma, and Jörg Ott. Security and privacy in Device-to-Device (D2D) communication: A review. *IEEE Communications Surveys and Tutorials*, 19(2):1054–1079, 2017.

[103] Jay Heiser and Mark Nicolett. Assessing the security risks of cloud computing. `https://www.gartner.com/doc/685308/assessing-security-risks-cloud-computing`, June 2008. Accessed: 2019-06-01.

[104] Constance L. Heitmeyer, Myla Archer, Elizabeth I. Leonard, and John McLean. Applying formal methods to a certifiably secure software system. *IEEE Trans. Software Eng.*, 34(1):82–98, 2008.

[105] Ronald Hes and John Borking, editors. *Privacy-Enhancing Technologies: The Path to Anonymity – Revised Edition*. Registratiekamer, 2000.

[106] Alan R. Hevner. The three cycle view of design science. *Scandinavian J. Inf. Systems*, 19(2):4, 2007.

[107] Alan R. Hevner, Salvatore T. March, Jinsoo Park, and Sudha Ram. Design science in information systems research. *MIS Quarterly*, 28(1):75–105, 2004.

[108] Jaap-Henk Hoepman. Privacy design strategies - (Extended Abstract). In *ICT Systems Security and Privacy Protection - 29th IFIP TC 11 International Conference, SEC 2014, Marrakech, Morocco, June 2-4, 2014. Proceedings*, pages 446–459, 2014.

[109] Patrick Hoffmann. UMLsec-Modellierung eines Zugriffskontrollmechanismus für Cloud-Umgebungen. Bachelor thesis, Technical University of Dortmund, June 2015.

[110] Jason I. Hong, Jennifer D. Ng, Scott Lederer, and James A. Landay. Privacy risk models for designing privacy-sensitive ubiquitous computing systems.

In *Proceedings of the Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques, Cambridge, MA, USA, August 1-4, 2004*, pages 91–100, 2004.

[111] Siv Hilde Houmb, Geri Georg, Jan Jürjens, and Robert France. An integrated security verification and security solution design trade-off analysis approach. In H. Mouratidis, editor, *Integrating Security and Software Engineering: Advances and Future Vision*, pages 190–219. Idea Group, 2006. Invited chapter.

[112] Hongxin Hu, Gail-Joon Ahn, and Jan Jorgensen. Detecting and resolving privacy conflicts for collaborative data sharing in online social networks. In *Twenty-Seventh Annual Computer Security Applications Conference, ACSAC 2011, Orlando, FL, USA, 5-9 December 2011*, pages 103–112, 2011.

[113] Giovanni Iachello and Jason I. Hong. End-user privacy in human-computer interaction. *Foundations and Trends in Human-Computer Interaction*, 1(1):1–137, 2007.

[114] Industrial Data Space Association. Industrial Data Space Use Case Broschuere. `https://www.internationaldataspaces.org/publications/use-case-brochure-2019-hannover-fair-edition/`. Accessed: 2019-06-01.

[115] International Association of Administrative Professionals (IAAP). IAPP-EY Annual Privacy Governance Report. Technical report, 2018. Accessed: 2019-06-01.

[116] International Function Point Users Group 2004. Function point counting practices manual - Release 4.2, 2004.

[117] Shareeful Islam, Haralambos Mouratidis, and Jan Jürjens. A framework to support alignment of secure software engineering with legal regulations. *Software and System Modeling*, 10(3):369–394, 2011.

[118] Shareeful Islam, Moussa Ouedraogo, Christos Kalloniatis, Haralambos Mouratidis, and Stefanos Gritzalis. Assurance of security and privacy requirements for cloud deployment model. *IEEE Transactions on Cloud Computing*, 2017. ISSN 2168-7161.

[119] ISO/IEC 27001. Information technology - Security techniques -Information security management systems - Requirements. Standard, International Organization for Standardization, Geneva, Switzerland, October 2013.

[120] ISO/IEC 27002. Information technology - Security techniques - Code of practice for information security management. Standard, International Organization for Standardization, Geneva, Switzerland, 2005.

[121] ISO/IEC 27005. Information technology - Security techniques -Information security risk management. Standard, International Organization for Standardization, Geneva, Switzerland, 2008.

[122] ISO/IEC 31000. Risk management - Principles and guidelines. Standard, International Organization for Standardization, Geneva, Switzerland, 2009.

[123] Thomas Jech. *Set theory, Third Edition*. Springer Monographs in Mathematics. Springer, 2002. ISBN 3-540-44085-2.

[124] Xin Jin, Ravi S. Sandhu, and Ram Krishnan. RABAC: Role-centric attribute-based access control. In *Computer Network Security - 6th International Conference on Mathematical Methods, Models and Architectures for Computer Network Security, MMM-ACNS 2012, St. Petersburg, Russia, October 17-19, 2012. Proceedings*, pages 84–96, 2012.

[125] Jan Jürjens. Secure information flow for concurrent processes. In *CONCUR 2000 - Concurrency Theory, 11th International Conference, University Park, PA, USA, August 22-25, 2000, Proceedings*, pages 395–409, 2000.

[126] Jan Jürjens. Modelling audit security for smart-card payment schemes with UMLsec. In *16th International Conference on Information Security (IFIPSEC"01)*, pages 93–108. IFIP, Kluwer, 2001.

[127] Jan Jürjens. Model-based security engineering with UML. In *Foundations of Security Analysis and Design III, FOSAD 2004/2005 Tutorial Lectures*, pages 42–77, 2004.

[128] Jan Jürjens. *Secure systems development with UML*. Springer, 2005. ISBN 978-3-540-00701-2.

[129] Jan Jürjens and Guido Wimmel. Formally testing fail-safety of electronic purse protocols. In *16th IEEE International Conference on Automated Software Engineering (ASE 2001), 26-29 November 2001, Coronado Island, San Diego, CA, USA*, pages 408–411, 2001.

[130] Jan Jürjens and Guido Wimmel. Security modelling for electronic commerce: The common electronic purse specifications. In *Towards The E-Society: E-Commerce, E-Business, and E-Government, The First IFIP Conference on E-Commerce, E-Business, E-Government (I3E 2001), October 3-5, Zürich, Switzerland*, pages 489–505, 2001.

[131] Christos Kalloniatis, Evangelia Kavakli, and Stefanos Gritzalis. Addressing privacy requirements in system design: the PriS method. *Requir. Eng.*, 13(3): 241–255, 2008.

[132] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2012, Bristol, UK, September 24-28, 2012. Proceedings, Part II*, pages 35–50, 2012.

[133] Kyo C. Kang, Sholom G. Cohen, James A. Hess, William E. Novak, and A. Spencer Peterson. Feature-oriented domain analysis (FODA) feasibility study. Technical report, Carnegie-Mellon University Software Engineering Institute, November 1990.

[134] Christian Kästner, Thomas Thüm, Gunter Saake, Janet Feigenspan, Thomas Leich, Fabian Wielgorz, and Sven Apel. Featureide: A tool framework for feature-oriented software development. In *31st International Conference on Software Engineering, ICSE 2009, May 16-24, 2009, Vancouver, Canada, Proceedings*, pages 611–614, 2009.

[135] Florian Kerschbaum. *Privacy-Preserving Computation*, pages 41–54. Springer Berlin Heidelberg, 2014. ISBN 978-3-642-54069-1.

[136] Jörg Kienzle, Wisam Al Abed, Franck Fleurey, Jean-Marc Jézéquel, and Jacques Klein. Aspect-oriented design with reusable aspect models. *Trans. Aspect-Oriented Software Development*, 7:272–320, 2010.

[137] Fabian Knirsch, Dominik Engel, Christian Neureiter, Marc Frîncu, and Viktor K. Prasanna. Model-driven privacy assessment in the smart grid. In *ICISSP 2015 - Proceedings of the 1st International Conference on Information Systems Security and Privacy, ESEO, Angers, Loire Valley, France, 9-11 February, 2015.*, pages 173–181, 2015.

[138] Karsten Krämer. Das Internet of Things und seine Plattformen. Bachelor's thesis, University of Koblenz Landau, July 2017.

[139] Kevin Lano, David Clark, and Kelly Androutsopoulos. Safety and security analysis of object-oriented models. In *Computer Safety, Reliability and Security, 21st International Conference, SAFECOMP 2002, Catania, Italy, September 10-13, 2002, Proceedings*, pages 82–93, 2002.

[140] Luigi Lavazza, Vieri Del Bianco, and Carla Garavaglia. Model-based functional size measurement. In *Proceedings of the Second International Symposium on Empirical Software Engineering and Measurement, ESEM 2008, October 9-10, 2008, Kaiserslautern, Germany*, pages 100–109, 2008.

[141] Torsten Lodderstedt, David A. Basin, and Jürgen Doser. SecureUML: A UML-based modeling language for model-driven security. In *UML 2002 - The Unified Modeling Language, 5th International Conference, Dresden, Germany, September 30 - October 4, 2002, Proceedings*, pages 426–441, 2002.

[142] Gabriel Maldoff. Top 10 operational impacts of the GDPR: Part 8 - pseudonymization. `https://iapp.org/news/a/top-10-operational-impacts-of-the-gdpr-part-8-pseudonymization/`, 2016. Accessed: 2019-06-01.

[143] Rene Meis and Maritta Heisel. Systematic identification of information flows from requirements to support privacy impact assessments. In *ICSOFT-PT 2015 - Proceedings of the 10th International Conference on Software Paradigm Trends, Colmar, Alsace, France, 20-22 July, 2015.*, pages 43–52, 2015.

[144] Rene Meis and Maritta Heisel. Supporting privacy impact assessments using problem-based privacy analysis. In *Software Technologies - 10th International Joint Conference, ICSOFT 2015, Colmar, France, July 20-22, 2015, Revised Selected Papers*, pages 79–98, 2015.

[145] Peter Mell and Timothy Grance. The NIST Definition of Cloud Computing. Technical report, National Institute for Standards and Technology, September 2011.

[146] Adam Moore. Defining privacy. *Journal of Social Philosophy*, 39(3):411–428, 2008. doi: 10.1111/j.1467-9833.2008.00433.x.

[147] Brice Morin, Jacques Klein, Olivier Barais, and Jean-Marc Jézéquel. A generic weaver for supporting product lines. In *Proceedings of the 13th International Workshop on Early Aspects*, EA '08, pages 11–18, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-032-6.

[148] Djedjiga Mouheb, Dima Alhadidi, Mariam Nouh, Mourad Debbabi, Lingyu Wang, and Makan Pourzandi. Aspect weaving in UML activity diagrams: A semantic and algorithmic framework. In *Formal Aspects of Component Software - 7th International Workshop, FACS 2010, Guimarães, Portugal, October 14-16, 2010, Revised Selected Papers*, pages 182–199, 2010.

[149] Haralambos Mouratidis, Paolo Giorgini, and Gordon A. Manson. Modelling secure multiagent systems. In *The Second International Joint Conference on Autonomous Agents & Multiagent Systems, AAMAS 2003, July 14-18, 2003, Melbourne, Victoria, Australia, Proceedings*, pages 859–866, 2003.

[150] James B. Nation. *Notes on Lattice Theory*, volume 60 of *Cambridge studies in advanced mathematics*. Cambridge University Press, 1998.

[151] National Institute of Standards and Technology. Security and privacy controls for Federal Information Systems and Organization, 2013.

[152] Phu Hong Nguyen, Koen Yskout, Thomas Heyman, Jacques Klein, Riccardo Scandariato, and Yves Le Traon. SoSPa: A system of security design patterns

for systematically engineering secure systems. In *18th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems, MoDELS 2015, Ottawa, ON, Canada, September 30 - October 2, 2015*, pages 246–255, 2015.

[153] Qun Ni, Dan Lin, Elisa Bertino, and Jorge Lobo. Conditional privacy-aware role based access control. In *Computer Security - ESORICS 2007, 12th European Symposium On Research In Computer Security, Dresden, Germany, September 24-26, 2007, Proceedings*, pages 72–89, 2007.

[154] NIST. *NIST Special Publication 800-53 Revision 4 Recommended Security Controls for Federal Information Systems and Organizations*. CreateSpace, Paramount, CA, 2012. ISBN 1470100363, 9781470100360.

[155] Hendrik J. G. Oberholzer and Martin S. Olivier. Privacy contracts as an extension of privacy policies. In *Proceedings of the 21st International Conference on Data Engineering Workshops, ICDE 2005, 5-8 April 2005, Tokyo, Japan*, page 1192, 2005.

[156] Object Management Group. Object Constraint Language - Version 2.4. Technical Report formal/2014-02-03, 2014.

[157] Object Management Group (OMG). UML Unified Modeling Language 2.5.1, 2017.

[158] Marie Caroline Oetzel and Sarah Spiekermann. A systematic methodology for privacy impact assessments: A design science approach. *EJIS*, 23(2):126–150, 2014.

[159] Marie Caroline Oetzel, Sarah Spiekermann, Ingrid Grüning, Harald Kelter, and Sabine Mull. Privacy impact assessment guideline for RFID applications. Technical report, Bundesamt für Sicherheit in der Informationstechnik, 2011.

[160] Institute of Distributed Systems. EU Privacy Patterns. `https://privacypatterns.eu/`. Accessed: 2019-06-01.

[161] Ian Oliver. Experiences in the development and usage of a privacy requirements framework. In *24th IEEE International Requirements Engineering Conference, RE 2016, Beijing, China, September 12-16, 2016*, pages 293–302, 2016.

[162] Organisation for Economic Co operation and Development. The OECD privacy framework, 2013.

[163] Boris Otto, Jan Jürjens, Jochen Schon, Sören Auer, Nadja Menz, Sven Wenzel, and Jan Cirullies. Industrial Data Space: Digital sovereignity over data. Technical report, Fraunhofer, 2016.

[164] Elda Paja, Fabiano Dalpiaz, Mauro Poggianella, Pierluigi Roberti, and Paolo Giorgini. STS-tool: Socio-technical security requirements through social commitments. In *2012 20th IEEE International Requirements Engineering Conference (RE), Chicago, IL, USA, September 24-28, 2012*, pages 331–332, 2012.

[165] Elda Paja, Fabiano Dalpiaz, and Paolo Giorgini. Modelling and reasoning about security requirements in socio-technical systems. *Data Knowl. Eng.*, 98: 123–143, 2015.

[166] Michalis Pavlidis and Shareeful Islam. SecTro: A CASE tool for modelling security in requirements engineering using secure tropos. In *Proceedings of the CAiSE Forum 2011, London, UK, June 22-24, 2011*, pages 89–96, 2011.

[167] Michalis Pavlidis, Shareeful Islam, Haralambos Mouratidis, and Paul Kearney. Modeling trust relationships for developing trustworthy information systems. *IJISMD*, 5(1):25–48, 2014.

[168] Siani Pearson. Taking account of privacy when designing cloud computing services. In *Proceedings of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing*, CLOUD '09, pages 44–52, Washington, DC, USA, 2009. IEEE Computer Society. ISBN 978-1-4244-3713-9.

[169] Siani Pearson and Damien Allison. A model-based privacy compliance checker. *IJEBR*, 5(2):63–83, 2009.

[170] Siani Pearson and Yun Shen. Context-aware privacy design pattern selection. In *Trust, Privacy and Security in Digital Business, 7th International Conference, TrustBus 2010, Bilbao, Spain, August 30-31, 2010. Proceedings*, pages 69–80, 2010.

[171] Pille Pullonen, Raimundas Matulevicius, and Dan Bogdanov. PE-BPMN: Privacy-enhanced business process model and notation. In *Business Process Management - 15th International Conference, BPM 2017, Barcelona, Spain, September 10-15, 2017, Proceedings*, pages 40–56, 2017.

[172] Nafees Qamar, Yves Ledru, and Akram Idani. Evaluating RBAC supported techniques and their validation and verification. In *Sixth International Conference on Availability, Reliability and Security, ARES 2011, Vienna, Austria, August 22-26, 2011*, pages 734–739, 2011.

[173] Fausto Rabitti, Elisa Bertino, Won Kim, and Darrell Woelk. A model of authorization for next-generation database systems. *ACM Trans. Database Syst.*, 16(1):88–131, 1991.

[174] Qusai Ramadan, Amir Shayan Ahmadian, Daniel Strüber, Jan Jürjens, and Steffen Staab. Model-based discrimination analysis: a position paper. In *Proceedings of the International Workshop on Software Fairness, FairWare@ICSE 2018, Gothenburg, Sweden, May 29, 2018*, pages 22–28, 2018.

[175] Dennis M. Riehle, Sven Jannaber, Patrick Delfmann, Oliver Thomas, and Jörg Becker. Automatically annotating business process models with ontology concepts at design-time. In *Advances in Conceptual Modeling - ER 2017 Workshops AHA, MoBiD, MREBA, OntoCom, and QMMQ, Valencia, Spain, November 6-9, 2017, Proceedings*, pages 177–186, 2017.

[176] Martin Rost. Datenschutz in 3D - Daten, Prozesse und Schutzziele in einem Modell. *Datenschutz und Datensicherheit*, 35(5):351–354, 2011.

[177] Martin Rost and Andreas Pfitzmann. Datenschutz-Schutzziele - revisited. *Datenschutz und Datensicherheit*, 33(6):353–358, 2009.

[178] Thomas Ruhroth and Jan Jürjens. Supporting security assurance in the context of evolution: Modular modeling and analysis with UMLsec. In *14th International IEEE Symposium on High-Assurance Systems Engineering, HASE 2012, Omaha, NE, USA, October 25-27, 2012*, pages 177–184, 2012.

[179] Karsten Saller, Malte Lochau, and Ingo Reimund. Context-aware DSPLs: Model-based runtime adaptation for resource-constrained systems. In *17th International Software Product Line Conference co-located workshops, SPLC 2013 workshops, Tokyo, Japan - August 26 - 30, 2013*, pages 106–113, 2013.

[180] Ravi S. Sandhu. Lattice-based access control models. *IEEE Computer*, 26(11): 9–19, 1993.

[181] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein, and Charles E. Youman. Role-based access control models. *IEEE Computer*, 29(2):38–47, 1996.

[182] Kurt Schneider, Eric Knauss, Siv Houmb, Dhri Sslam, and Jan Jürjens. Enhancing security requirements engineering by organisational learning. *Requirements Engineering Journal (REJ)*, 17(1):35–56, 2012.

[183] Markus Schumacher, Eduardo B. Fernández-Buglioni, Duane Hybertson, Frank Buschmann, and Peter Sommerlad. *Security Patterns - Integrating Security and Systems Engineering*. Wiley, 2005. ISBN 978-0-470-85884-4.

[184] Mary Shaw. The coming-of-age of software architecture research. In *Proceedings of the 23rd International Conference on Software Engineering, ICSE 2001, 12-19 May 2001, Toronto, Ontario, Canada*, pages 656–664, 2001.

[185] Till Shümmer. The public privacy - patterns for filtering personal information in collaborative systems. In *Proceedings of CHI workshop on Human-Computer-Human-Interaction Patterns, 2004.*

[186] Daniel J. Solove. A taxonomy of privacy. *University of Pennsylvania Law Review*, 154(3):477–560, Januar 2006.

[187] Samaneh Soltani, Mohsen Asadi, Dragan Gasevic, Marek Hatala, and Ebrahim Bagheri. Automated planning for feature model configuration based on functional and non-functional requirements. In *16th International Software Product Line Conference, SPLC '12, Salvador, Brazil - September 2-7, 2012, Volume 1*, pages 56–65, 2012.

[188] Sarah Spiekermann. The challenges of privacy by design. *Commun. ACM*, 55 (7):38–40, 2012.

[189] Sarah Spiekermann and Lorrie Faith Cranor. Engineering privacy. *IEEE Trans. Software Eng.*, 35(1):67–82, 2009.

[190] Sarah Spiekermann, Alessandro Acquisti, Rainer Böhme, and Kai-Lung Hui. The challenges of personal data markets and privacy. *Electronic Markets*, 25 (2):161–167, 2015.

[191] David Steinberg, Frank Budinsky, Marcelo Paternostro, and Ed Merks. *EMF: Eclipse Modeling Framework 2.0*. Addison-Wesley Professional, 2nd edition, 2009. ISBN 0321331885.

[192] Harald Störrle. Semantics and verification of data flow in UML 2.0 activities. *Electr. Notes Theor. Comput. Sci.*, 127(4):35–52, 2005.

[193] Harald Störrle. How are conceptual models used in industrial software development? A descriptive survey. In *Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering, EASE 2017, Karlskrona, Sweden, June 15-16, 2017*, pages 160–169, 2017.

[194] Theeraporn Suphakul and Twittie Senivongse. Development of privacy design patterns based on privacy principles and UML. In *18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, SNPD 2017, Kanazawa, Japan, June 26-28, 2017*, pages 369–375, 2017.

[195] ClouDAT Project Team. The final documentation of the ClouDAT project. Technical report, 2015.

[196] The European Commission. Commission Recommendation of 10 October 2014 on the Data Protection Impact Assessment Template for Smart Grid and Smart Metering Systems . *Official Journal of the European Union*, L 300/63, 2014.

[197] The European Parliament and the Council of the Europeam Union. Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data. *Official Journal of the European Union*, L 281, 1995.

[198] The European Parliament and the Council of the Europeam Union. Regulation (EU) 2016/679 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data. *Official Journal of the European Union*, L 119, 2016.

[199] The United States Department of Justice. The privacy act of 1974. Technical report, 1974.

[200] Slim Trabelsi, Gregory Neven, and Dave Raggett. Report on design and implementation of the PrimeLife Policy Language and engine. Technical report, 2011.

[201] G. W. van Blarkom, John J. Borking, and Eddy Olk. Handbook of privacy and privacy-enhancing technologies: The case of intelligent software agents. Technical report, Privacy Incorporated Software Agent Consortium, Den Haag, 2003.

[202] Wynand van Staden and Martin S. Olivier. Using purpose lattices to facilitate customisation of privacy agreements. In *Trust, Privacy and Security in Digital Business, 4th International Conference, TrustBus 2007, Regensburg, Germany, September 3-7, 2007, Proceedings*, pages 201–209, 2007.

[203] Wynand van Staden and Martin S. Olivier. On compound purposes and compound reasons for enabling privacy. *J. UCS*, 17(3):426–450, 2011.

[204] Wynand J C van Staden and Martin S Olivier. Purpose organisation. In *Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA2005)*, Sandton, South Africa, 6 2005. Research in progress paper, published electronically.

[205] Sarah Vetter. Cyber security aspects of the Industrial Data Space for Zero-Defect-Manufacturing. Master's thesis, University of Koblenz Landau, June 2019.

[206] Zhiguo Wan, Jun-e Liu, and Robert H. Deng. HASBE: A hierarchical attribute-based solution for flexible and scalable access control in cloud computing. *IEEE Trans. Information Forensics and Security*, 7(2):743–754, 2012.

[207] David Wright and Paul De Hert. *Privacy Impact Assessment*. Springer Netherlands, 2012.

[208] David Wright and Kush Wadhwa. Introducing a privacy impact assessment policy in the EU member states. *International Data Privacy Law*, 3:13, 2013.

[209] David Wright, Kush Wadhwa, Paul De Hert, and Dariusz Kloza. A Privacy Impact Assessment Framework for Data Protection and Privacy Rights. Technical report, 2011.

[210] Tianshui Wu and Gang Zhao. A novel risk assessment model for privacy security in internet of things. *Wuhan University Journal of Natural Sciences*, 19 (5):398–404, Oct 2014. ISSN 1993-4998.

[211] Kim Wuyts, Riccardo Scandariato, Bart De Decker, and Wouter Joosen. Linking privacy solutions to developer goals. In *Proceedings of The Forth International Conference on Availability, Reliability and Security, ARES 2009, March 16-19, 2009, Fukuoka, Japan*, pages 847–852, 2009.

[212] Kwangsun Yoon and Ching-lai Hwang. *Multiple Attribute Decision Making: An Introduction*. SAGE University Paper series on Quantitative Applications in the Social Sciences, 1995. ISBN 0-8039-5486-7.

[213] Christian Zinke, Jürgen Anke, Kyrill Meyer, and Johannes Schmidt. Modeling, analysis and control of personal data to ensure data privacy – A use case driven approach. In Denise Nicholson, editor, *Advances in Human Factors in Cybersecurity*, pages 87–96. Springer International Publishing, 2018.

Typeset February 3, 2020