Chapter 5 Technical tools for designing data protection

5.1 Introductory remarks

This Chapter is dedicated to a more applied perspective in the technological domain. As explained above, one of the main challenges faced by PbD, and now by DPbD, is finding a proactive approach that combines the legal and technical perspectives to design privacy or data protection. The task of identifying technologies that protect rights (and principles) must not be limited to legislators[1699]. Anyone who develops or uses information technology to process data should take legal rules into account by adopting organisational and technological solutions that promote those rules[1700].

Thus, the present Chapter investigates the existing technical tools and methods for designing data protection. It first introduces some general systems and software engineering concepts. Then it focuses on privacy engineering approaches, by looking at some significant contributions for PbD and DPbD, and at the risk assessment framework, which is crucial for Article 25 of the GDPR.

Given the e-health care sector, and the case study on EHR, the Chapter then presents some suitable PETs and recognised international standards that are useful for EHR implementation. These insights are tools for defining the DPbD guidelines to be applied in the EHR environment.

5.2 System and software development design

The EHR system is complex, and has a set of components that includes both hardware and software: database management systems and their hardware, EHR software with its architecture and interface, and the net-

1699 Giovanni Sartor. *L'informatica giuridica e le tecnologie dell'informazione: Corso di informatica giuridica*. Vol. 2. G. Giappichelli Editore, 2016. ISBN: 9788892105935, p. 41.

1700 *See ibid.*, which refers to "values" instead of rules from a legal informatics perspective.

work[1701]. This section deals briefly with systems engineering aspects and secondly with software development issues.

Generally, a system is built through the interdisciplinary approach of systems engineering[1702]. System development mainly involves three different implementations: infrastructure, platform design and software design[1703]. So, systems engineering is not merely software development.

System requirements (i.e. its properties) are defined in the early development stage in order to select the specific architectures and technologies solutions to be built. In particular, functional requirements determine how the system behaves and interacts, what capabilities it provides and what information it processes[1704]. The non-functional requirements refer to the criteria required to understand how well the functions of the system are achieved, such as effectiveness, quality and cost. The definition of system requirements follows the identification of stakeholders' requirements, which are statements of what experts, users, customers, and personnel need from the specific system to be implemented[1705]. While system requirements are defined in formal or semi-formal language, component requirements can be expressed as textual and problem-oriented requirements, and through use cases.

So, privacy or data protection needs may be identified by the stakeholders who then provide the requirements to the developers to take them into account while defining the system requirements. Actually, PbD and DPbD demands the translation of rules into design requirements both in hardware and in software[1706].

The integration of privacy rules may raise terminological problems since some terms are used in the legal field with different meanings than the

1701 *See* as an example the openEHR technical specifications available at <specifications.openehr.org/>. Last accessed 06/10/2021. In particular, Figure 7 describes the health service environment with multiple layers and components.

1702 For an introduction to system engineering *see* the first chapter of Bruce Powel Douglass. *Agile Systems Engineering*. Online version. Morgan Kaufmann, 2016. ISBN: 9780128023495. In this book, systems engineering is defined as "an interdisciplinary approach to building complex and technologically diverse systems".

1703 *See* e.g. the life cycle in Douglass, *op. cit.*, p. 22.

1704 Douglass, *op. cit.*, p. 5.

1705 On whom may be the stakeholders *see* Douglass, *op. cit.*, p. 68.

1706 For PbD *see* Ann Cavoukian, Stuart Shapiro, and R. Jason Cronk. "Privacy engineering: Proactively embedding privacy, by design". In: *Office of the Information and Privacy Commissioner* (2014).

same terms have in the technological domain[1707]. As discussed above, privacy and data protection principles are expressed in broader terms than engineering requirements are, and are subject to interpretation[1708]. Technology operates by on-off rules, whereas law by interpretative rules[1709].

Therefore, legal rules should be analysed, requirements or use cases may be identified, and then they may be translated into concrete functional or non-functional system requirements by following a methodology[1710]. Some rules may affect the entire architecture of an information system, while others may regulate its run-time level[1711].

Moreover, as previously noted, the adoption of a particular concept of privacy or data protection configures different frameworks of values and dimensions[1712]. Incorporating values requires the competence of a system designer, but also comprehensive knowledge of the legal field or the support of other legal experts[1713]. Taking into account data protection needs

1707 *See* Stefan Schiffner et al. "Towards a roadmap for privacy technologies and the General Data Protection Regulation: A transatlantic initiative". In: *Privacy Technologies and Policy. 6th Annual Privacy Forum, APF 2018*. Springer. 2018, pp. 24–42, p. 35.

1708 *See* Chapter 2, Section 2.3. *See* also Alshammari and Simpson, "Towards a principled approach for engineering privacy by design", pp. 163–164.

1709 *See* Waldman, "Privacy's Law of Design", p. 1257.

1710 *See* N. Van Dijk et al. "Right engineering? The redesign of privacy and personal data protection". In: *International Review of Law, Computers & Technology* 32.2 – 3 (2018), pp. 230–256, pp. 239–241, which reports the opinions of representatives from the engineering community. Some experts are critical of the ability to translate legal principles, whereas others are more optimistic. Following a methodology really contributes to the effort.

1711 *See* Koops and Leenes, "Privacy regulation cannot be hardcoded. A critical comment on the 'privacy by design' provision in data-protection law", p. 164. The authors classify Article 17 of the DPD as a system level requirement, and the time for data retention as a run-time requirement. They even classify language requirements as "requirements for the policy language that derive from legal provisions".

1712 A summary of the different frameworks and rationales is provided by Tamó-Larrieux, *Designing for privacy and its legal framework: data protection by design and default for the internet of things*, pp. 27–39.

1713 On the complexity of achieving technical design that incorporates values *see* Mary Flanagan, Daniel C. Howe, and Helen Nissenbaum. "Embodying values in technology: Theory and practice". In: *Information technology and moral philosophy*. Cambridge University Press, 2008, pp. 322–353. ISBN: 9780511498725. *See* also Chapter 2, Section 2.3.

is not a trivial problem. A privacy system engineering methodology should be adopted[1714].

An EHR system also embeds a software system. Software development is a well-structured activity, which includes multiple phases and interactions[1715]. Software development can follow different methodologies.

Methodologies can be divided into two main categories: structured methodologies, which collect models with detailed planning, management and documentation, and agile methodologies, which are characterised by iterative processes and less planning[1716].

To explain software development in relation to PbD, ENISA uses the waterfall model, which can be considered a structured methodology that includes the seven following phases: concept development, analysis, design, implementation, testing, evaluation, and maintenance[1717]. The waterfall model is a traditional development model that relies on documentation and detailed planning and management[1718]. Each phase may rely on a privacy engineering approach[1719]. The various stages and their implementation are sequential, meaning that one phase must not be started before the previous has ended and has been documented[1720]. The advantage of the waterfall model seems to be the great attention to the first phase on concept development and identifying requirements. Since it is not easy to go back to a previous phase, each one should be carefully carried out.

1714 Privacy engineering approaches will be presented in the next Section 5.3.

1715 *See* Sartor, *L'informatica giuridica e le tecnologie dell'informazione: Corso di informatica giuridica*, pp. 114–117.

1716 Hans-Christian Estler et al. "Agile vs. structured distributed software development: A case study". In: *Empirical Software Engineering* 19.5 (2014), pp. 1197–1224, which tries to compare the models in a case study.

1717 *See* Danezis et al., *Privacy and Data Protection by design – from policy to engineering*, p. 18.

1718 *See* Seda Gürses and Joris Van Hoboken. "Privacy after the agile turn". In: *The Cambridge Handbook of Consumer Privacy*. Cambridge University Press, 2018, pp. 579–601. ISBN: 9781316831960, p. 582.

1719 Danezis et al., *Privacy and Data Protection by design – from policy to engineering*, p. 17: "To support privacy by design throughout the software development each of these phases rely on different concepts. In the concept development and analysis phases so called privacy design strategies (defined further on) are necessary. The known concept of a design pattern is useful during the design phase, whereas concrete (privacy-enhancing) technologies can only be applied during the implementation phase".

1720 *See* Olga Filipova and Rui Vilão. *Software Development From A to Z*. Springer, 2018. ISBN: 9781484239445, p. 27, which reports as phases: requirements, analysis, design, coding, testing, and maintenance.

As a result, data protection requirements may be cautiously taken into account with the waterfall model. At the same time, the disadvantage seems to be that this methodology is not very flexible and takes a long time to carry out, and if a data protection requirement is not considered in the first phase, it will be difficult and expensive to change the final version of the project later on[1721]. It has been pointed out that the waterfall cycle is lacking the creative process that is needed for PbD[1722]. So, this methodology may be used for DPbD implementation, but presents some challenges.

In addition to the waterfall model, over the last few decades the agile software model has been increasingly adopted[1723]. It has been reported that it seems to be the mainstream software development method worldwide[1724]. The agile model is "based on iterative development, frequent inspection and adaptation, and incremental deliveries in which requirements and solutions evolve through collaboration in cross-functional teams and through continuous stakeholder feedback"[1725]. Hence, this model is characterised by short development cycles, continuous testing, simplicity and user centricity[1726]. The development usually follows the modularity principle, which allows independent implementation of modules in the system to manage its complexity[1727]. Developers can continuously add new features or modify existing ones in a never-ending development phase which is called *perpetual beta*[1728]. A large number of approaches can be identified as agile methods[1729].

1721 *See* the comment in Filipova and Vilão, *op. cit.*, p. 28.

1722 *See* Schiffner et al., "Towards a roadmap for privacy technologies and the General Data Protection Regulation: A transatlantic initiative", p. 39.

1723 *See* Gürses and Van Hoboken, "Privacy after the agile turn", pp. 582–583.

1724 Rashina Hoda, Norsaremah Salleh, and John Grundy. "The rise and evolution of agile software development". In: *IEEE software* 35.5 (2018), pp. 58–63.

1725 ISO/IEC/IEEE. *ISO/IEC/IEEE 26515:2018 Systems and software engineering — Developing information for users in an agile environment*. Tech. rep. ISO/IEC/IEEE Second edition 2018–12, 2018.

1726 *See* Gürses and Van Hoboken, "Privacy after the agile turn", p. 582.

1727 *See* Gürses and Van Hoboken, *op. cit.*, p. 586.

1728 *See* Gürses and Van Hoboken, *op. cit.*, p. 593.

1729 *See* David Parsons. "Agile software development methodology, an ontological analysis". In: <www. researchgate.net/> (2011), which refers to Agile Microsoft Solutions Framework, Agile UP, Crystal Clear, DSDM, eXtreme Programming (XP), Feature Driven Development, Scrum. This article contains a useful ontology of agile methods which tries to show the common elements.

Despite the potential risk of infringements in a continuous process, it is possible to quickly redesign features on demand. Changing requirements even late in development is one of the 12 principles of the "Manifesto for Agile Software Development" of 2001[1730]. This Manifesto has been criticised for being too vague for a scientific work, but it started the discussion on how to use an iterative development method[1731]. The methodology focuses on solving problems, rather than following fixed planning[1732]. Agile planning is dynamic and employs continuous verification and incremental progress. In fact, agile often involves planning only for the short term and the implementation of processes goes in parallel[1733]. The iterative development cycle is still based on requirements and feedback.

The advantage of agile methods seems to be the ability to quickly change the requirements at any phase with an interdisciplinary team. As a result, DPbD technical implementation remains an ongoing process as

---

1730 *See* Kent Beck et al. *Manifesto for agile software development*. <agilemanifesto.org/>. 2001. The principles are: "1) Our highest priority is to satisfy the customer through early and continuous delivery of valuable software; 2) Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage; 3) Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale; 4) Business people and developers must work together daily throughout the project; 5) Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done; 6) The most efficient and effective method of conveying information to and within a development team is face-to-face conversation; 7) Working software is the primary measure of progress; 8) Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely; 9) Continuous attention to technical excellence and good design enhances agility; 10) Simplicity -the art of maximizing the amount of work not done- is essential; 11) The best architectures, requirements, and designs emerge from self-organizing teams; 12) At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly". It is worth noting that these principles pay great attention to good design and teamwork, even promoting a sort of interdisciplinarity in principle 4.

1731 *See* Maarit Laanti, Jouni Similä, and Pekka Abrahamsson. "Definitions of agile software development and agility". In: *European Conference on Software Process Improvement*. Springer. 2013, pp. 247–258, which reports criticism and provides a table on agile principles and what they emphasise.

1732 Douglass, *Agile Systems Engineering*, p. 44. The book summarises the benefits at p. 83.

1733 ISO/IEC/IEEE, *ISO/IEC/IEEE 26515:2018 Systems and software engineering — Developing information for users in an agile environment*.

required by law. At the same time, the disadvantage seems to be that this methodology does not take into account the need to carefully plan the requirements before the first delivery of the project, with all the potential risks for data protection[1734]. It has been argued that while agility requires sprints, privacy analysis needs time and patience[1735]. So, once again this methodology may be used for DPbD implementation, but it also presents some challenges. The requirement and planning phase should remain a relevant stage for DPbD, within the possibility of changing the *status quo* pursuant to a new rule or a new aspect of the data processing.

In 2017, the Norwegian Data Protection Authority released some guidelines on "software development with Data Protection by Design and by Default"[1736]. The Authority declared that it had used as starting points the Microsoft Security Development Lifecycle (SDL), the Secure Software Development Life Cycle (S-SDLC) and the ENISA report *Privacy and Data Protection By design – from policy to engineering*[1737]. The guidelines contained a circular diagram with seven key activities in the software development process as pieces of a ring puzzle. This circularity represents the ongoing process needed to apply data protection by design and aims to show a general methodology for its development.

The authority described seven activities or steps: training, requirements, design, coding, testing, release, and maintenance. Within an organisation, the description of these activities may be summarised as follows:

– Training: the management and employees of an organisation should have knowledge of which data protection requirements are applicable, which information security tools are usable and which methodology

1734 *See* the comment in Filipova and Vilão, *Software Development From A to Z*, p. 28.

1735 *See* Schiffner et al., "Towards a roadmap for privacy technologies and the General Data Protection Regulation: A transatlantic initiative", p. 36.

1736 *See* Datatilsynet Norwegian Data Protection Authority. *Guidelines on software development with Data protection by Design and by Default*. 2017. According to Bygrave, these guidelines are useful for the application of Article 25 of the GDPR. *See* Bygrave, "Chapter IV Controller and Processor (Articles 24–43). Article 25. Data protection by design and by default", p. 577. This document was also quoted by the EDPB in European Data Protection Board, *Guidelines 4/2019 on Article 25 Data Protection by Design and by Default*. As argued in Bincoletto, "European Union – EDPB Guidelines 4/2019 on Data Protection by Design and by Default", p. 578, the guidelines of the Norwegian DPA are a valuable knowledge base for engineering data protection and building-in the requirements of the GDPR.

1737 Danezis et al., *Privacy and Data Protection by design – from policy to engineering*.

should be applied. To achieve this know-how, a training plan should be prepared by the organisation;
- Requirements: data protection and information security product and operational requirements should be established in advance for the development team in order to mitigate the possible risks. These requirements are strictly related to the concrete context and the applicable legal framework. Moreover, they could be expressed as a checklist and follow international standards. In this step, a risk assessment and, if required, a DPIA should be performed;
- Design: all previous specifications should be reflected in the design step, when the organisation should set the design requirements describing software characteristics and functionality. Two categories could be identified. Firstly, the so-called "data oriented design requirements" are: minimising the amount of personal data; hiding and protecting the collected data; separating the processing or the storage; aggregating the data as much as possible; and configuring data protection by default settings. Secondly, the "process oriented design requirements" are: providing information on how the software works and data are processed; giving control to the data subject; documenting all the adopted technical safeguards and demonstrating compliance with the rules[1738];
- Coding: the aim of this activity is "to write secure code", which is regularly subject to code analysis and code reviews. Developers should use recognised and up-to-date tools for software development from a list approved by the organisation and should document every adopted choice. All of the code functions and modules should be safe, even if they are developed by third parties;
- Testing: in this activity the implementation is compared with the planned data protection and security requirements by testers. In particular, security, dynamic, fuzz, and penetration testing should be performed;
- Release: an incident response plan should be prepared in the release phase;
- Maintenance: handling incidents and data breaches as planned is important, as well as maintaining a management system for data protection and information security.

The approach recommended by the Norwegian authority is particularly interesting for DPbD since it includes a strong analysis of the applicable

1738 These requirements follow the "privacy design strategies" that will be presented *infra* in Section 5.3.2.

legal framework and risk assessment before the design stage, it considers the difference between "data-oriented design requirements" and "process-oriented design requirements", which respectively refer to technical and organisational requirements, and it is convincing on the need to adopt an interdisciplinary approach[1739].

Any approach should take into account the personal data life cycle since data are processed both in the system and in the software. Tamó-Larrieux groups the possible life cycle phases into four main steps: data collection, data analysis, the use of data, data erasure or deletion[1740]. This author classifies the planning process and accessing and retrieving activities during the collection phase. The analysis step refers to storing, mining and managing databases, while the use step includes making predictions and decisions. The last phase identifies the moment when data is erased or recycled for further use.

Personal data life cycle may be re-classified as "data collection", "data use" *in latu sensu* and "data erasure". The phases are relevant for the data protection domain since different rules, and then measures, apply in each of them[1741]. Another valuable distinction is considering data at rest, data in use, and data in transit. While defining the requirements for the design stage, all these distinctions should be taken into account[1742].

After these brief considerations on system and software development, the following section will investigate the privacy engineering approaches.

5.3 Overview of privacy engineering approaches

In 1967 privacy appeared for the first time as research topic in a computer science conference[1743]. In the 1980s, David Chaum proposed cryptographic protocols to control and monitor data exchange that combined system

1739 This categorisation will be taken into account in the next Chapter for the set of guidelines.

1740 *See* the life cycle of data framework in Tamó-Larrieux, *Designing for privacy and its legal framework: data protection by design and default for the internet of things*, pp. 149–151.

1741 Tamó-Larrieux argued that legislators have the data life cycle in mind while establishing the data protection framework. *See* Tamó-Larrieux, *op. cit.*, p. 151.

1742 Even these distinctions will be used in the next Chapter for the set of guidelines.

1743 Tamó-Larrieux, *Designing for privacy and its legal framework: data protection by design and default for the internet of things*, p. 104.

requirements with privacy[1744]. Over the 1990s a privacy technology community grew rapidly[1745]. At that time privacy conversations were mainly focused on preserving internet anonymity[1746].

As mentioned in Chapter 2, in the 1990s engineers started developing privacy-enhancing technologies to customise some information flow rules through technical design, while protecting privacy[1747]. PETs are ICT measures, applications or tools, that address a single dimension of privacy, such as anonymity or confidentiality, by eliminating or minimising personal data or by preventing unlawful uses without losing the functionality of an information system[1748]. So, PETs were progressively developed for the preservation of multiple values, including confidentiality, anonymity, transparency and control[1749]. As an example, confidentiality may be en-

1744 *See* David Chaum. "Untraceable electronic mail, return addresses, and digital pseudonyms". In: *Communications of the ACM* 24.2 (1981), pp. 84–90 and David Chaum. "Showing credentials without identification". In: *Workshop on the Theory and Application of Cryptographic Techniques*. Springer. 1985, pp. 241–244, which briefly describes the basic credential system.

1745 *See* George Danezis and Seda Gürses. "A critical review of 10 years of privacy technology". In: *Proceed- ings of surveillance cultures: a global surveillance society* (2010), pp. 1–16, p. 1, which reports the history. Some valuable studies from that period are: Victoria Bellotti and Abigail Sellen. "Design for privacy in ubiquitous computing environments". In: *Proceedings of the Third European Conference on Computer-Supported Cooperative Work 13–17 September 1993, Milan, Italy ECSCW'93*. Springer. 1993, pp. 77–92; Simon G Davies. "Re-engineering the right to privacy: how privacy has been transformed from a right to a commodity". In: *Technology and privacy: The new landscape* 143 (1997), pp. 143–166 Philip E. Agre and Marc Rotenberg. *Technology and privacy: The new landscape*. Mit Press, 1998. ISBN: 9780262011624.

1746 In 1993 the New Yorker published a famous cartoon by Peter Steiner where a dog sitting on a chair at a desk in front of a computer says to another dog sitting on the floor: "On the Internet, nobody knows you're a dog".

1747 *See* Chapter 2, Section 2.3. *See* also Reidenberg, "Lex informatica: The formulation of information policy rules through technology"; Bygrave, "Hardwiring privacy"; Van Rossum, Gardeniers, et al., *Privacy-enhancing technologies: The path to anonymity*.

1748 *See* Rubinstein, "Regulating privacy by design", p. 1411; Danezis and Gürses, "A critical review of 10 years of privacy technology"; European Commission, *Communication from the Commission to the European Parliament and the Council on Promoting Data Protection by Privacy Enhancing Technologies (PETs)*. For examples of technologies and techniques for enhancing trust *see* Le Métayer, "Whom to Trust? Using Technology to Enforce Privacy".

1749 *See* Bygrave, "Hardwiring privacy", p. 757.

forced with encryption, and security with an identity management system (IDMS)[1750].

The technologies for enforcing privacy have been classified into two main categories: "technologies for avoiding or reducing as much as possible the disclosure of personal data, hence enforcing the data minimisation principle" that avoid giving trust to data controllers (i.e. hard privacy), and "technologies for enforcing the rights of the subject if personal data is disclosed or processed", hence placing a certain amount of trust over controllers (i.e. soft privacy)[1751]. Thus, hard privacy is mostly about data minimisation seeking to avoid any disclosure, whereas soft privacy is mostly about data management seeking to share data in a way that protects and enforces rights[1752]. In the second category data management and users' choices play an important role.

The concept of PbD emerged with PET development and is strictly related to them since the approach of implementation can include these tools as building blocks[1753]. The same statement may refer to DPbD. PETs and standards may be components of a PbD or DPbD approach, but this concept is more comprehensive than a set of tools[1754]. Protecting personal data *by design* demands a proactive privacy engineering approach.

1750 *See* on encryption Le Métayer, "Whom to Trust? Using Technology to Enforce Privacy", p. 400; Whitfield Diffie and Susan Landau. *Privacy on the line: The politics of wiretapping and encryption*. Updated and expanded edition. The MIT Press, 2007. ISBN: 9780262042406; and on IDMS Danezis and Gürses, "A critical review of 10 years of privacy technology", p. 3.

1751 *See* Le Métayer, "Whom to Trust? Using Technology to Enforce Privacy", p. 397. According to the author, the use of these technologies is not sufficient, since a more proactive and comprehensive approach is necessary.

1752 *See* Rubinstein and Good, "The trouble with Article 25 (and how to fix it): the future of data protection by design and default", p. 9. As an example, hard privacy includes anonymous communication channels, selective disclosure credentials, private information retrieval, and homomorphic encryption. Soft privacy includes cookie management tools, privacy dashboards, and auditable secure logs.

1753 *See* once again Hustinx, "Privacy by design: delivering the promises"; Kroener and Wright, "A strategy for operationalizing privacy by design"; D'Acquisto et al., *Privacy by design in big data: an overview of privacy enhancing technologies in the era of big data analytics*; Tsormpatzoudi, Berendt, and Coudert, "Privacy by design: from research and policy to practice–the challenge of multi-disciplinarity"; Bygrave, "Hardwiring privacy".

1754 *See* Cavoukian, Shapiro, and Cronk, "Privacy engineering: Proactively embedding privacy, by design".

According to Gürses *et al.*, privacy engineering is "an emerging field of research that focuses on designing, implementing, adapting and evaluating theories, methods, techniques and tools to systematically capture and address privacy issues in the development of sociotechnical systems"[1755]. Privacy engineering mainly derives from the software engineering field, but it also embeds other computer science fields, including information security, human-computer interaction and machine learning[1756].

Privacy engineering means using engineering principles and processes to embed privacy and data protection features and measures in technical design on a case-by-case basis and for the data life-cycle[1757]. Actually, this computer science field may be used for all the following goals[1758]:

- "Designing and constructing processes, products, and systems with privacy in mind that appropriately collect or use personal information;
- Supporting the development, implementation, and measurement of privacy policies, standards, guidelines, and rules;
- Analysing software and hardware designs and implementation from a privacy and user experience perspective;
- Supporting privacy audits;
- Working with other stakeholders to ensure privacy requirements are met outside as well as inside the engineering space".

Regulation by design is aimed at the first goal primarily. Privacy or data protection requirements may turn into either functional components of the

1755 Gürses and Van Hoboken, "Privacy after the agile turn", p. 581.

1756 Van Dijk et al., "Right engineering? The redesign of privacy and personal data protection", p. 235.

1757 *See* Tamó-Larrieux, *Designing for privacy and its legal framework: data protection by design and default for the internet of things*, p. 232. *See* also the definition of Michelle Dennedy, Jonathan Fox, and Tom Finneran. *The privacy engineer's manifesto: getting from policy to code to QA to value*. Apress, 2014. ISBN: 9781430263562, p. 29: "Privacy engineering as a discrete discipline or field of inquiry and innovation may be defined as using engineering principles and processes to build controls and measures into processes, systems, components, and products that enable the authorized, fair, and legitimate processing of personal information". Interestingly, this book also specifies that "privacy engineering is not merely a call for mindful engineering where personal information is involved. The call for privacy engineering use and study is a call for leadership, innovation, and even a good measure of courage to change the status quo for design and information management". So, this discipline is even useful for technological innovation.

1758 Dennedy, Fox, and Finneran, *op. cit.*, p. 30.

system or non-functional ones[1759]. So, systematic methods should provide the means for representing, eliciting and analysing the requirements[1760].

In the literature, several approaches of privacy engineering can be distinguished[1761]. The approaches may define strategies and goals that developers should take into account when working on a concrete project or they may establish priorities and development methods.

First of all, the taxonomy of "privacy-by-policy" and "privacy-by-architecture" is frequently used for explaining privacy engineering approaches[1762]. The former concept refers to strategies that implement the "notice-and-choice" principle, while the latter refers to strategies that minimise the collection of information by using pseudonymisation or anonymisation techniques[1763]. However, it seems that this categorisation is mainly focused on US concepts. It may be argued that both HIPAA Privacy and Security Rules in the US and DPbD in the EU require more comprehensive and hybrid strategies.

Some approaches focus on modelling privacy requirements from an organisational point of view for adopting privacy by design. PbD is actually an approach that requires both technical and organisational measures. Lentzsch *et al.* observed a lack of adoption of PbD approaches focused on process-driven strategies and socio-technical design[1764]. So, they proposed a socio-technical design (STD) approach that brought together users, privacy experts and developers through workshops and used a modelling annota-

---

1759 Cavoukian stated that privacy is usually ancillary to the primary purposes of a system. Then, it is frequently a non-functional requirement. *See* Cavoukian, Shapiro, and Cronk, "Privacy engineering: Proactively embedding privacy, by design".

1760 *See* Guarda and Zannone, "Towards the development of privacy-aware systems", p. 19.

1761 *See* the overviews by Seda Gürses and Jose M. Del Alamo. "Privacy engineering: Shaping an emerging field of research and practice". In: *IEEE Security & Privacy* 14.2 (2016), pp. 40–46; Sarah Spiekermann and Lorrie Faith Cranor. "Engineering privacy". In: *IEEE Transactions on software engineering* 35.1 (2008), pp. 67–82; Guarda and Zannone, "Towards the development of privacy-aware systems".

1762 *See* e.g. Spiekermann and Cranor, "Engineering privacy", p. 73; Cavoukian, Shapiro, and Cronk, "Privacy engineering: Proactively embedding privacy, by design", pp. 12–13; Gürses and Del Alamo, "Privacy engineering: Shaping an emerging field of research and practice".

1763 Spiekermann and Cranor, "Engineering privacy", p. 79.

1764 Christopher Lentzsch et al. "Integrating a Practice Perspective to Privacy by Design". In: *International Conference on Human Aspects of Information Security, Privacy, and Trust*. Springer. 2017, pp. 691–702.

tion called SeeMe. Their modelling is guided by questions addressed to the participants and further aspects should be added according to the discussion[1765].

The PriS method is a requirement engineering methodology, but it proposes to incorporate privacy requirements as organisational goals to be achieved in the early development stage[1766]. PriS uses eight privacy goals, namely "identification, authentication, authorisation, data protection, anonymity, pseudonymity, unlinkability and unobservability". The method first requires eliciting the goals that are relevant for the concrete project. Then, it is necessary to analyse the impact of the selected goals on business processes and their support systems and to model the privacy-related processes with the Enterprise Knowledge Development (EKD) framework[1767]. After that, the developer can identify the techniques that support these privacy-related processes with privacy-process patterns. The PriS approach is also based on a formal representation of the phases[1768]. Despite the complexity and comprehensiveness of this approach, it does not specifically take into account privacy or data protection principles as defined by the law. However, new approaches use the PriS methodology to create new privacy process patterns that are useful for engineers[1769].

In a prominent study investigating how "engineering privacy by design" could be addressed, Gürses *et al.* defined five steps that have to be re-iterated many times when developing a system with privacy and data minimisation embedded at the core[1770]:

1. Clearly describing system functionality (i.e. functional requirements analysis);
2. Minimising data (e.g. using advanced cryptography techniques);
3. Modelling attackers, threats and risks, including a typical risk analysis;

1765 Lentzsch et al., *op. cit.*

1766 Christos Kalloniatis, Evangelia Kavakli, and Stefanos Gritzalis. "Addressing privacy requirements in system design: the PriS method". In: *Requirements Engineering* 13.3 (2008), pp. 241–255; Christos Kalloniatis, Petros Belsis, and Stefanos Gritzalis. "A soft computing approach for privacy requirements engineering: The PriS framework". In: *Applied Soft Computing* 11.7 (2011), pp. 4341–4348.

1767 *See* Kalloniatis, Kavakli, and Gritzalis, "Addressing privacy requirements in system design: the PriS method", p. 245.

1768 *See* Kalloniatis, Kavakli, and Gritzalis, *op. cit.*, pp. 247–249.

1769 *See* Vasiliki Diamantopoulou et al. "Supporting privacy by design using privacy process patterns". In: *IFIP International Conference on ICT Systems Security and Privacy Protection*. Springer. 2017, pp. 491–505.

1770 *See* Gürses, Troncoso, and Diaz, "Engineering privacy by design", pp. 18–19.

4. Analysing multilateral security requirements since privacy measures should not be detrimental to other important security objectives of a system;
5. Implementing and testing the design to understand whether it embeds the solution "that fulfils the integrity requirements revealing the minimal amount of private data".

According to this study, data minimisation has a central role in the PbD approach, and it shall be considered its guiding principle. Article 25 of the GDPR highlights the importance of this principle by using it as an example of the data protection principle. At the same time, Gürses' approach included security and risk assessment as fundamental steps from a privacy engineering point of view.

A group of researchers proposed a methodology for enabling PbD in medical record sharing[1771]. As a methodology, the CHINO project proposed starting with the extraction of compliance and business requirements from the legal provisions and the involved stakeholders, respectively, by following five steps with different actors[1772]:

1. Identification of business requirements, which is performed by a chief information officer;
2. Identification of compliance requirements, which is performed by a chief compliance officer;
3. Definition of compliance-aware data management scenarios, which is performed by a business analyst;
4. Definition of executable processes and policies, which is performed by a business analyst and by developers;
5. Deployment and execution inside run-time environment, which are performed by developers.

This approach used both European and HIPAA rules for extracting requirements that are applicable to a specific use case in the healthcare domain. The requirements have been identified as "privacy policies", and they take into account different roles. The benefit of this study is showing how requirements and data management operations can be modelled by using the Business Process Model and Notation (BPMN)[1773].

1771 Jovan Stevovic et al. "Enabling privacy by design in medical records sharing". In: *Reforming European Data Protection Law*. Springer, 2015, pp. 385–406. ISBN: 9789401793858.

1772 Stevovic et al., *op. cit.*

1773 *See* the current BPMN specifications at <www.bpmn.org>. Last accessed 06/10/2021.

In the *Preliminary Opinion on privacy by design* the EDPS quoted the framework of so-called "Six protection goals for privacy engineering" as an example of existing useful methodologies[1774]. This framework was proposed by Hansen *et al.* in 2015 and it defined six goals that can be used by engineers for deriving requirements, choosing techniques and technologies, and evaluating the privacy impacts and conditions of systems[1775]. Three goals are the CIAD triad, i.e. confidentiality, integrity and availability. These traditional security principles are fundamental for any development of ICT system[1776].

Beyond these goals, according to this framework, engineers should consider another triad: unlinkability, transparency and intervenability[1777]. The goal of unlinkability entails that "processes have to be operated in such a way that the privacy-relevant data are not linkable to any privacy-relevant information outside of the domain"[1778]. This goal embeds the principles of data minimisation and purpose limitation, and it can be achieved through pseudonymisation or anonymisation. In this study transparency refers to openness and accountability and it means that "all privacy-relevant data processing – including the legal, technical, and organizational setting – can be understood and reconstructed at any time"[1779]. Logging, detailed documentation, and information delivery mechanisms are common techniques for achieving transparency. Finally, the research defines intervenability as the "property that intervention is possible concerning all ongoing or planned privacy-relevant data processing", including the execution of data

1774 European Data Protection Supervisor, *Opinion 5/2018, Preliminary Opinion on privacy by design*, p. 13.

1775 Marit Hansen, Meiko Jensen, and Martin Rost. "Protection goals for privacy engineering". In: *2015 IEEE Security and Privacy Workshops*. IEEE. 2015, pp. 159–166.

1776 Engineers may use encryption, access control mechanisms, and other techniques like redundancy and virtualisation.

1777 This triad has also been endorsed by the Spanish DPA in the Guide on privacy by design. The authority created a table where the triad is associated with the GDPR's principles: unlinkability embeds data minimisation, storage limitation, and integrity and confidentiality; transparency embeds lawfulness, fairness and transparency, and purpose limitation; intervenability/control embeds purpose limitation, accuracy, integrity and confidentiality, and accountability. *See* Agencia Espanõla de Protección de Datos, *A Guide to Privacy by Design*, pp. 13–14.

1778 Hansen, Jensen, and Rost, "Protection goals for privacy engineering", p. 160.

1779 *Ibid.*

subject's rights[1780]. Overall, the six goals may conflict with one another and then the developer may mitigate such a conflict by deciding on concrete priorities[1781]. This approach is an abstract model that is useful for guiding the developer by using strategies, but these strategies are still quite broad, and they do not define explicit requirements.

Another approach quoted by the EDPS is the "privacy design patterns" framework. In general, design patterns are tools used for making decisions about the organisation of a software system since they describe its commonly recurring structure and components[1782]. It has been highlighted that the work on privacy patterns is recommended in the field of PbD[1783]. In fact, detailed privacy patterns could be used for deciding how system architecture should be implemented in specific parts. These patterns have been classified by the literature, and they include several PETs[1784]. Thanks to an international and institutional collaboration, the portal *privacypatterns.eu* collects and discusses the published privacy patterns[1785]. As an example, the "Pseudonymous Messaging" pattern establishes that "a messaging service is enhanced by using a trusted third party to exchange the identifiers of the communication partners by pseudonyms"[1786]. A standardisation process may enhance the use of design patterns. As such, the approach is not comprehensive, and it is very abstract. So, privacy design

1780 *Ibid.* As regards this last goal, the authors states that few techniques could have been implemented.

1781 Hansen, Jensen, and Rost, *op. cit.*, p. 161.

1782 Danezis et al., *Privacy and Data Protection by design – from policy to engineering*, p. 17; Jaap-Henk Hoepman. "Privacy design strategies". In: *IFIP International Information Security Conference*. Springer, 2014, pp. 446–459, p. 448.

1783 Koot and Laat, "Privacy from an Informatics Perspective", p. 246; Agencia Espanõla de Protección de Dados, *A Guide to Privacy by Design*.

1784 *See* Munawar Hafiz. "A collection of privacy design patterns". In: *Proceedings of the 2006 conference on Pattern languages of programs*. 2006, pp. 1–13; Munawar Hafiz. "A pattern language for developing privacy enhancing technologies". In: *Software: Practice and Experience* 43.7 (2013), pp. 769–787; Jörg Lenhard, Lothar Fritsch, and Sebastian Herold. "A literature study on privacy patterns research". In: *2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE. 2017, pp. 194–201. A long selection on patterns is also provided by Agencia Espanõla de Protección de Dados, *A Guide to Privacy by Design*, pp. 32–43.

1785 *See* the official website at <privacypatterns.eu>. Last accessed 06/10/2021.

1786 *See* the pattern at <privacypatterns.eu/#/patterns/pseudonymous-messaging>. Last accessed 06/10/2021.

patterns should be used with other design strategies and architectural tactics[1787].

Privacy is considered both a functional and a non-functional requirement in the "Privacy- Enhancing ARchitectures" (PEARs) methodology. The PEARs framework is based on the analysis of quality attributes of a system and it proposes four tactics for achieving privacy protection through requirements[1788]. The developer first analyses and identifies the scenarios, selects architecture techniques that influence the scenarios (i.e. tactics) and verifies the impact of the techniques on response measures[1789]. The four tactics for privacy by design that influence the non-functional requirements of a system are classified as minimisation tactics (e.g. anonymisation), enforcement tactics (e.g. access rights), accountability tactics (e.g. logging), and modifiability tactics (e.g. change policies)[1790]. These tactics are described with patterns, and they use PETs. So, the approach proposes a methodology that includes both the use of patterns or PETs and the description of non-functional requirements.

In 2017, Guarda *et al.* proposed a methodology based on three building blocks for applying privacy and data protection at the beginning of the design process, for solving the problem of the natural language of the legal requirements, and for providing evidence on the compliance checking[1791]. Firstly, they elaborated "a declarative framework to specify the processing of data for certain purposes together with legal requirements and security policies at design-time"[1792]. Secondly, they introduced an interdisciplinary approach for deriving formal specifications from legal rules. Thirdly, they suggested automated techniques to solve security analysis and compliance checking problems. This interdisciplinary research was based on data protection requirements of the DPD.

1787 *See* Hoepman, "Privacy design strategies".

1788 *See* Antonio Kung. "PEARs: privacy enhancing architectures". In: *Proceedings of the Annual Privacy Forum of 2014*. Springer. 2014, pp. 18–29.

1789 *See* Kung, *op. cit.*, p. 21.

1790 *See* Kung, *op. cit.*, pp. 23–24.

1791 *See* Paolo Guarda, Silvio Ranise, and Hari Siswantoro. "Security analysis and legal compliance checking for the design of privacy-friendly information systems". In: *Proceedings of the 22nd ACM on Symposium on Access Control Models and Technologies*. 2017, pp. 247–254.

1792 Guarda, Ranise, and Siswantoro, *op. cit.*, p. 248.

As regards the formal representation of legal norms, the great contribution of the legal informatics field should be mentioned[1793]. It does not propose engineering approaches, but it provides valuable instruments to be taken into account. In particular, to represent legal resources the so-called LegalRuleML, a robust and expressive XML annotation, created a framework for modelling normative rules that satisfies the legal domain requirements[1794]. LegalRuleML provided an integrated and self-contained representation of legal resources available on the Web that is useful for a legal reasoning level combined with an ontological layer. As previously mentioned, the Akoma-Ntoso standard also provided the schema for the structure and the semantic components of digital legislative documents in machine readable form[1795]. It has been pointed out that LegalRuleML can represent and store the logical content of the legal provisions, while Akoma-Ntoso can be used to tag the original textual content of the legal documents[1796]. The DAPRECO (DAta Protection REgulation COmpliance) research project used these instruments and the legal ontology PrOnto[1797],

1793 On legal informatics *see* Giovanni Sartor, Maria Angela Biasiotti, and Fabrizio Turchi. *Tecnologie e abilità informatiche per il diritto*. G. Giappichelli Editore, 2018. ISBN: 9788834839409; Sartor, *L'informatica giuridica e le tecnologie dell'informazione: Corso di informatica giuridica*; Giovanni Sartor. "Il diritto nell'informatica giuridica". In: *Rivista di filosofia del diritto* 4 Speciale (2015), pp. 71–92; Massimo Durante and Ugo Pagallo. *Manuale di informatica giuridica e diritto delle nuove tecnologie*. Utet Giuridica, 2012. ISBN: 9788859807773; Giovanni Sartor. "Legislative information and the web". In: *Legislative XML for the Semantic Web*. Springer, 2011, pp. 11–20; Mariangela Biasiotti et al. "Legal informatics and management of legislative documents". In: *Global Center for ICT in Parliament Working Paper* 2 (2008); Vittorio Frosini and Donato Antonio Limone. *L'insegnamento dell'informatica giuridica*. Liguori, 1990. ISBN: 8820719169.

1794 *See* Monica Palmirani et al. "LegalRuleML: XML-based rules and norms". In: *International Workshop on Rules and Rule Markup Languages for the Semantic Web*. Springer. 2011, pp. 298–312; Tara Athan et al. "LegalRuleML: Design principles and foundations". In: *Reasoning Web International Summer School*. Springer. 2015, pp. 151–188.

1795 *See* Palmirani and Vitali, "Akoma-Ntoso for legal documents"; Palmirani, "Legislative change management with Akoma-Ntoso".

1796 Livio Robaldo et al. "Formalizing GDPR provisions in Reified I/O logic: the DAPRECO knowledge base". In: *Journal of Logic, Language and Information* (2019), pp. 1–49.

1797 Monica Palmirani et al. "PrOnto: Privacy ontology for legal reasoning". In: *International Conference on Electronic Government and the Information Systems Perspective*. Springer. 2018, pp. 139–152; Palmirani et al., "Legal Ontology for Modelling GDPR Concepts and Norms"; Palmirani et al., "PrOnto Ontology

to create a knowledge base on the GDPR that is useful for legal reasoning and automated compliance checking[1798].

Overall, engineering approaches have attempted to provide more guidance to developers on privacy by design. The research to date has tended to focus on PbD and privacy strategies trying to combine system engineering methods and modelling with broad concepts and principles. Three other relevant approaches for engineering privacy are the "PRIPARE project", "privacy design strategies" and the "LIDDUN methodology", which will be analysed separately in the following subsections.

5.3.1 The PRIPARE project

The PEARs project was connected to another EU-funded project called "Preparing Industry to PbD by supporting its Application in Research" (PRIPARE)[1799]. At the time of this project the GDPR was under discussion, so the legislation used by the team was its draft version of 2015.

PRIPARE's methodology included the typical system engineering phases – namely analysis, design, implementation, verification, release, maintenance and decommission – and it added the central phase "environment & infrastructure", which required the implementation of an appropriate organisational structure during the application of all the other steps[1800]. In spite of the indication of these phases, the PRIPARE methodology is iterative and non-linear[1801]. Several roles should be involved in the

---

Refinement Through Open Knowledge Extraction". On other privacy legal ontologies *see* Leone, Di Caro, and Villata, "Taking stock of legal ontologies: a feature-based comparative analysis"; Oliveira Rodrigues et al., "Legal ontologies over time: a systematic mapping study". *See* also Chapter 2, Section 2.3.

1798 Robaldo et al., "Formalizing GDPR provisions in Reified I/O logic: the DAPRECO knowledge base".

1799 *See* Nicolás Notario et al. "PRIPARE: a new vision on engineering privacy and security by design". In: *Cyber Security and Privacy Forum*. Springer. 2014, pp. 65–76; Nicolás Notario et al. "PRIPARE: integrating privacy best practices into a privacy engineering methodology". In: *2015 IEEE Security and Privacy Workshops*. IEEE. 2015, pp. 151–158; Nicolás Notario et al. *PRIPARE. Privacy-and Security-by design Methodology Handbook. 2016*. 2017.

1800 Notario et al., *op. cit.*, p. 14.

1801 The report specified that the PRIPARE methodology is compatible with most agile methodologies since the seven phases can be reiterated many times. *See* Notario et al., *op. cit.*, pp. 103–104.

development process: systems engineers, privacy and security officers, data subjects, DPAs, end users and project managers.

During the analysis phase, given a set of privacy and security principles obtained with a legal assessment, the requirements gathering of PRIPARE should be performed with the involvement of all stakeholders and an initial risk assessment. The principles used by PRIPARE were: "consent and choice; purpose legitimacy and specification; collection limitation; data minimization; use retention and disclosure limitation; accuracy and quality; openness, transparency and notice; individual participation and access; accountability; information security; privacy compliance"[1802]. These principles refer both to FIPs, OECD Guidelines and GDPR principles. For each principle a fixed list of goal-oriented guidelines should be mapped and then techniques to fulfil these guidelines should be identified.

As a result, operational requirements are obtained from privacy principles. For example, guidelines of the data minimisation principles are: "avoid and minimise the use of personal data along its whole life-cycle"; "limit the ability of external parties from inferring personal data from sources coming from different controllers"; "minimize the traces left by transactions and interactions with a system or service"[1803].

Having defined the operational requirements, the design phase should concretely build the system through privacy and security patterns, tactics, PEARs, strategies and PETs. So, this approach took into account different architecture approaches during the effective implementation. This project also showed that the implementation of privacy by design should follow the high-level analysis of the legal principles and the operationalisation of these principles in guidelines and strategies. For this reason, privacy experts should be given a seat at the table.

The PRIPARE project then described several formal approaches for architecture design and classified existing techniques from the literature[1804]. In order to check whether the implementation respects legal requirements, the system developer and the project manager should express the implementation with formal semantics and use a verification tool or a theorem prover to verify the implementation with the properties and the scenarios[1805]. Prior to the release, even a dynamic analysis on the code should be

1802 Notario et al., *op. cit.*, p. 40.
1803 *See* Notario et al., *op. cit.*, p. 43.
1804 *See* Notario et al., *op. cit.*, pp. 56–62.
1805 *See* Notario et al., *op. cit.*, pp. 67–68.

performed through testing tools, instrumentation techniques, and dynamic flow analysis[1806].

After the release of the system, an incident response plan should be created, and the privacy impact assessment should be published. Examination and re-examination should be iterative phases during the use of the system, including periodical risk assessment, and every analysis should be reported and documented in detail to ensure accountability.

This project provided a list of guidelines and applied criteria that are associated with privacy principles[1807]. These guidelines and the PRIPARE method may be considered a useful starting point for a DPbD approach. It should be noted, however, that a DPbD implementation should now take into account the data protection principles and requirements of the approved text of the GDPR.

An interesting project that is using GDPR concepts and lexicon is the "Architectural View for Data Protection by Design" of KU Leuven University[1808]. This research provides a meta-model for the data protection architectural viewpoint with UML class diagrams[1809]. The model identifies GDPR actors, their roles in the processing activities, and provides data flow diagrams (DFDs) and some requirements expressed as criteria (e.g. the documentation criterion). Interestingly, the research has been validated with a case study on the e-health domain[1810].

5.3.2 Privacy design strategies

Privacy design strategies are general strategies that are aimed at achieving privacy protection by limiting how the system structure is realised during the first phases of the development cycle[1811]. The strategies should guide the software development cycle in the concept and analysis phase in choosing quality attributes. So, in this approach privacy influences non-functional requirements. Later, in the design phase design patterns

---

1806 *See* Notario et al., *op. cit.*, p. 69.
1807 *See* Notario et al., *op. cit.*, pp. 120–132.
1808 *See* Laurens Sion et al. "An architectural view for data protection by design". In: *2019 IEEE International Conference on Software Architecture (ICSA)*. IEEE. 2019, pp. 11–20.
1809 *See* Sion et al., *op. cit.*, p. 14.
1810 The research in Sion et al., *op. cit.* refers to a patient monitoring system.
1811 Danezis et al., *Privacy and Data Protection by design – from policy to engineering*, p. 18.

remain useful, as do PETs during the implementation phase. These strategies usually suggest a waterfall methodology, but they simply refer to the requirement phase that is useful in agile methods, too[1812].

A key study on privacy design strategies was carried out by Hoepman in 2014[1813]. In particular, eight privacy design strategies were proposed with their respective design patterns. The data protection rules used by this framework were the OECD Guidelines, Article 8 of the European Convention of Human Rights, and the DPD[1814]. So, the selected principles were: "purpose limitation (comprising both specification of the purpose and limiting the use to that stated purpose); data minimisation; data quality; transparency (openness in OECD terms); data subject rights (in terms of consent, and the right to view, erase, and rectify personal data); the right to be forgotten; adequate protection (security safeguards in OECD terms); data portability; data breach notifications; accountability and (provable) compliance"[1815]. It may be noted that these principles follow both the OECD Guidelines, the FIPs and European principles (e.g. right to be forgotten and data portability).

The first four strategies were data-oriented, while the other four were process-oriented. The strategies can be summarised as follows[1816]:

1. Minimise. The first strategy states that the amount of personal data should be limited to the minimum. Minimising the amount of data means selecting data before collection, or anonymising (and pseudonymising) data after collection. Thus, this strategy corresponds to the data minimisation principle under the GDPR or the "minimum necessary rule" of the HIPAA, and to purpose limitation;
2. Hide. This strategy requires hiding personal data from anybody or from unauthorised entities preserving data confidentiality. Typical examples of hide design patterns are encryption and anonymisation that achieve data minimisation;
3. Separate. The third strategy is aimed at processing personal data in a distributed way whenever possible by separating the performed activities or the data storage related to a single individual. Decentralised

1812 Jaap-Henk Hoepman. "Privacy Design Strategies (The Little Blue Book)". In: *Radboud University Repository* (2018), p. 22.
1813 *See* Hoepman, "Privacy design strategies".
1814 Hoepman, *op. cit.*, pp. 449–450.
1815 Hoepman, *op. cit.*, p. 451.
1816 *See* Hoepman, *op. cit.*; Danezis et al., *Privacy and Data Protection by design – from policy to engineering*; Agencia Espanõla de Protección de Datos, *A Guide to Privacy by Design*, pp. 16–24.

services or separation of databases are useful for this strategy to respect the purpose limitation principle;
4. Aggregate, later defined as Abstract. The last data-oriented strategy requires processing personal data at the highest level of aggregation that corresponds to the least level of detail that is useful to the controller. Again, anonymisation techniques may be appropriate;
5. Inform. As the first process-oriented strategy, informing data subjects on the existence and context of the processing is highly important for protecting transparency and data subject's rights. The information should refer to the purpose and means of the processing, including the security of the used system and documentation on design. The data subject should be informed of the recipients and existing rights. Design patterns of this strategy are: platforms for privacy preferences, data breach notification, and transparency-enhancing techniques;
6. Control. According to this strategy the data subject should have the means to control the processing of personal data. As an example, user-centric identity management helps the individual control the processed data. The principles for this strategy are data quality and data portability;
7. Enforce. This strategy states that a privacy policy should be in place. Actually, the strategy refers to practices and measures compatible with the legal requirements, instead of referring to the concrete document where the information is provided. So, this strategy is strictly related to the accountability principle;
8. Demonstrate. Even this last strategy is connected to accountability. The controller should demonstrate compliance with the applicable legal requirements. Logging and auditing are typical examples of techniques for this strategy.

This framework later took into account the GDPR requirements and assigned applicable architectural tactics to the privacy strategies[1817]. This resulted in a more concrete approach. At the same time, Hoepman *et al.* used the FTC's version of the FIPs to include the US market and the concept of PII. As an example, the tactics for the "minimize strategy" are: "exclude", meaning refraining from processing partly or entirely with opt-out solutions; "select", meaning deciding on the full or partial use of personal data with opt-in-solutions; "strip", meaning removing unnecessary

1817 *See* Michael Colesky, Jaap-Henk Hoepman, and Christiaan Hillen. "A critical analysis of privacy design strategies". In: *2016 IEEE Security and Privacy Workshops (SPW)*. IEEE. 2016, pp. 33–40.

personal data categories in the system; and "destroy", meaning deleting personal data after the retention period[1818].

In addition to strategies and tactics, several examples of state of the art techniques and technologies were classified in Hoepman's Little Blue Book in 2018. This collection should address organisations, designers, and engineers that need to build privacy by design systems[1819]. Privacy design strategies are useful for defining requirements, but they should be combined with the applicable privacy and data protection principles. Besides, anonymisation is not always feasible.

5.3.3 LIDDUN methodology

The last methodology of this overview is the LIDDUN methodology, which is based on the creation and analysis of the system data flows and of privacy threat patterns[1820]. In particular, LIDDUN is based on diagrams for mapping entities, processes and flows, and stresses the importance of risk analysis[1821].

The LIDDUN methodology has been recognised by the literature as a modelling framework that supports the elicitation of privacy requirements and mitigation of privacy threats[1822]. The acronym LIDDUN actually embeds the following privacy threat categories: "linkability, identifiability, non-repudiation, detectability, disclosure of information, unawareness,

1818 *See* Colesky, Hoepman, and Hillen, *op. cit.*, p. 35.

1819 *See* Hoepman, "Privacy Design Strategies (The Little Blue Book)".

1820 Danezis et al., *Privacy and Data Protection by design – from policy to engineering*, p. 13.

1821 *See* the comment of the EDPS in European Data Protection Supervisor, *Opinion 5/2018, Preliminary Opinion on privacy by design*, p. 14.

1822 *See* Mina Deng et al. "A privacy threat analysis framework: supporting the elicitation and fulfilment of privacy requirements". In: *Requirements Engineering* 16.1 (2011), pp. 3–32; Kim Wuyts, Riccardo Scandariato, and Wouter Joosen. "LIND(D)UN privacy threat tree catalog". In: *CW Reports 675* (2014); Kim Wuyts, Riccardo Scandariato, and Wouter Joosen. "Empirical evaluation of a privacy-focused threat modeling methodology". In: *Journal of Systems and Software* 96 (2014), pp. 122–138; Sion et al., "An architectural view for data protection by design", p. 12; Notario et al., *PRIPARE. Privacy-and Security-by design Methodology Handbook. 2016*; Laurens Sion et al. "Interaction-based privacy threat elicitation". In: *2018 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE. 2018, pp. 79–86.

non-compliance"[1823]. These threats may be posed by an external entity during a data flow where a user is performing a process.

The LIDDUN framework models the data flow, and provides threat tree catalogues for describing the envisaged scenarios of the same threats. The mapping of the privacy threats is combined with software-based system components and a formal modelling[1824]. This modelling may help the developer elicit concrete privacy requirements and select technical solutions that are able to fulfil these requirements.

Hence, unlike the PRIPARE methodology and privacy strategies that start with the analysis of principles or goals, and after that perform a risk analysis, LIDDUN begins with risk modelling and then includes the requirements. LIDDUN does not explain how to select the PETs that correspond to a privacy requirement, but it provides mitigation strategies and state of the art techniques based on the envisaged threats. It does not even use a specific set of privacy principles[1825]. The benefit of this approach is using semantics and abstract modelling to guide developers while recognising the risks. This approach is not comprehensive, but it may be used during a privacy impact assessment as a technical component[1826].

So far, this Chapter has presented several privacy engineering approaches. Overall, these frameworks should not be seen as self-excluding. During a risk assessment, data flow mapping and threat analysis and modelling like LIDDUN may help the developer identify risks and find solutions to mitigate these risks. During the system and software development, after choosing a development method (e.g. waterfall or agile), privacy design strategies or goals, design patterns, architectural tactics and PETs help the developer to define the functional and non-functional system requirements with privacy protection. A comprehensive methodology like PRIPARE provides guidelines for all the phases of the development life cycle and includes stakeholders' organisational and management level.

1823 The description of the threats is provided in Sion et al., *op. cit.*

1824 *See* Kristian Beckers. "Comparing privacy requirements engineering approaches". In: *2012 Seventh International Conference on Availability, Reliability and Security*. IEEE. 2012, pp. 574–581, p. 577.

1825 For this criticism of LIDDUN *see* Alshammari and Simpson, "Towards a principled approach for engineering privacy by design", pp. 165–166; and Maria Grazia Porcedda. "'Privacy by Design' in EU Law". In: *Privacy Technologies and Policy. 6th Annual Privacy Forum, APF 2018*. Springer. 2018, pp. 183–204, p. 189.

1826 Sion et al., "Interaction-based privacy threat elicitation", p. 85.

Risk analysis and assessment are pivotal components of all the methodologies. In fact, a privacy engineering framework should always be combined with a privacy risk analysis. The next section deals with this aspect, by investigating general concepts and discussing some applicable methodologies for the data protection impact assessment.

5.4 Guidance on the risk assessment framework

Privacy engineering and DPbD require an efficient approach to risk assessment. As mentioned in Chapter 2, risk is the product of likelihood of an event and its severity: *risk = likelihood × severity*.

Where risk may be defined as the "effect of uncertainty on objectives", likelihood is "the chance of something happening" – that is the event or "occurrence or change of a particular set of circumstances"[1827] – and severity is the measure of the possible consequences of the source of this event, i.e. its potential harm. So, the event or threat identifies a circumstance or set of circumstances that causes harm to personal data. The likelihood – i.e. the probability that this event will happen[1828] – is frequently scaled from 0 to 1, whereas the severity – i.e. the impact – is scaled with qualitative terms.

In the data protection domain, likelihood and severity are both usually scaled from "low", "medium", "high" to even "very high"[1829]. At the same time, scores 1, 2, 3 may be assigned to the three first levels. As regards the likelihood, if the event or threat is unlikely to happen, the level is low; if it is possible or likely to materialise, the level is respectively medium or high[1830]. Severity refers to the consequences of the event on the individual.

1827 ISO. *ISO/Guide 73:2009(en) Risk management — Vocabulary*. Tech. rep. ISO/TMBG, 2009.

1828 ISO, *op. cit.*, specifies that likelihood may refer to either probability or frequency. Actually, the word probability usually refers to the mathematical term. Therefore, ISO points out that "in risk management terminology, "likelihood" is used with the intent that it should have the same broad interpretation as the term "probability" has in many languages other than English".

1829 European Union Agency for Network & Information Security, *Handbook on Security of Personal Data Processing*; Fabio Guasconi et al. *Reinforcing trust and security in the area of electronic communications and online services. Sketching the notion of "state-of-the-art" for SMEs in security of personal data processing*. European Union Agency for Network and Information Security, 2018, p. 18.

1830 D'Acquisto and Panagopoulou, *Guidelines for SMEs on the security of personal data processing*, p. 29.

Where the individual may encounter few inconveniences, the level is low, whereas where the inconveniences are significant and serious, the level is high[1831]. This evaluation performed by the data controller is a qualitative process.

While discussing the security risk assessment of data processing, ENISA suggested considering separately the risks related to the network and the technical resources of the data controller, to processes and procedures of the data processing operations, to different parties and people involved in the data processing, and to the business sector and specific scale of the processing (e.g. large scale)[1832]. More specifically, the data controller should use as parameters for the processes and procedures of the data processing the category of personal data, the criticality of the processing operations (e.g. profiling), the volume of data, special characteristics of the data controller (e.g. public entity), and special characteristics of the data subjects (e.g. minors)[1833]. So, the data controller could assign to each mentioned area a level and a score to added up with the others[1834]. The security risk assessment may be carried out in parallel with a privacy or data protection risk assessment.

In sum, the data controller should evaluate likelihood and severity as "low, medium or high" and combine the levels to obtain the risk level.

1831 *See* all the descriptions of the levels in European Union Agency for Network & Information Security, *Handbook on Security of Personal Data Processing*, p. 11: "Low, individuals may encounter a few minor inconveniences, which they will overcome without any problem (time spent re-entering information, annoyances, irritations, etc.). Medium, individuals may encounter significant inconveniences, which they will be able to overcome despite a few difficulties (extra costs, denial of access to business services, fear, lack of understanding, stress, minor physical ailments, etc.). High, individuals may encounter significant consequences, which they should be able to overcome albeit with serious difficulties (misappropriation of funds, blacklisting by financial institutions, property damage, loss of employment, subpoena, worsening of health, etc.). Very high, individuals may encounter significant, or even irreversible consequences, which they may not overcome (inability to work, long-term psychological or physical ailments, death, etc.)".

1832 *See* European Union Agency for Network & Information Security, *op. cit.*, pp. 12–15; D'Acquisto and Panagopoulou, *Guidelines for SMEs on the security of personal data processing*, pp. 24–25.

1833 *See* D'Acquisto and Panagopoulou, *op. cit.*, p. 21.

1834 ENISA also provides an example of final range of 4–5 for low, 6–8 for medium and 9–12 for high. *See* the table in D'Acquisto and Panagopoulou, *op. cit.*, p. 31.

Thus, the level of risk may be visualised as reported in the following Table 5.1[1835].

Table 5.1 Risk level

Likelihood \ Severity	Low	Medium	High
Low	Low risk	Medium risk	High risk
Medium	Low risk	Medium risk	High risk
High	Medium risk	High risk	High risk

Having defined these fundamental concepts applicable to an assessment, it is worth examining how to conduct a data protection risk assessment, i.e. the DPIA. This task is complex since it requires several categories of skills, including risk management, business expertise and knowledge of security[1836].

As mentioned in Chapter 2, Section 2.5.2, Article 29 Working Party released some guidelines on DPIA and the GDPR[1837]. Valuable DPIA guidelines have also been provided by the European project PRIAM and the French DPA, the CNIL[1838].

1835 Own graphic inspired by: European Union Agency for Network & Information Security, *Handbook on Security of Personal Data Processing*; Guasconi et al., *Reinforcing trust and security in the area of electronic communications and online services. Sketching the notion of "state-of-the-art" for SMEs in security of personal data processing*, p. 14.

1836 Jules Sarrat and Raphael Brun. "DPIA: how to carry out one of the key principles of accountability". In: *Privacy Technologies and Policy. 6th Annual Privacy Forum, APF 2018*. Springer. 2018, pp. 172–182.

1837 Article 29 Working Party, *Guidelines on Data Protection Impact Assessment (DPIA) and determining whether processing is "likely to result in a high risk" for the purposes of Regulation 2016/679*.

1838 Other useful guidelines that are applicable outside the EU can be derived from the NIST risk management framework of the US government and from ISO/IEC standards. NIST publishes several guidelines on computer security and risk assessment. *See* the official website at <csrc.nist.gov/publications/>. Last accessed 06/10/2021. Noteworthy among them is the NIST Privacy Framework National Institute of Standards and NIST Technology. *NIST Privacy Framework: A Tool for Improving Privacy Through Enterprise Risk Management*,

The PRIAM framework combines the legal and technical fields to create a privacy risk assessment that is based on the specific attributes and components of a system[1839]. In fact, this approach starts with information gathering that collects information on the functional components of the system, the interface, the data flows, the supporting assets and the actors and roles (i.e. stakeholders). Even the technical and organisational measures already implemented should be analysed and collected as information. According to Le Métayer *et al.*, the assessment should involve the entire life cycle of the processing performed through a system[1840]. The identification of actors and roles is fundamental for defining data flows. PRIAM defines a risk source as "any entity (individual or organization) which may process (legally or illegally) data belonging to a data subject and whose actions may directly or indirectly, intentionally or unintentionally lead to privacy harms". Each risk source should be described through accurate attributes and be evaluated using a scale. The controller should also identify feared events and privacy harms. After this first phase, the risk assessment can be carried out following a methodology based on harm trees[1841]. As a result, the risk assessment is a systematic, traceable and computational activity[1842].

The CNIL approach has been recommended by the PRIPARE project[1843]. The methodology is divided into four steps[1844]:

1. Defining and describing the characteristics of data processing. During this phase the controller should identify the other subjects and the recipients of personal data, and this subject should also describe the operations and the supporting assets[1845]. Even the standards applicable to processing should be identified;

Version 1.0. National Institute of Standards and Technology, 2020. On risk analysis and HIPAA *see* Thompson, *Building a HIPAA-Compliant Cybersecurity Program*. The ISO standards will be quoted in the next Section.

1839 *See* Daniel Le Métayer and Sourya Joyee De. *PRIAM: a Privacy Risk Analysis Methodology*. Research Report RR-8876, Inria, Research Centre Grenoble, 2016.

1840 *See* Le Métayer and De, *op. cit.*, p. 9.

1841 *See* Le Métayer and De, *op. cit.*, pp. 32–38.

1842 Le Métayer and De, *op. cit.*, p. 40.

1843 *See* Notario et al., *PRIPARE. Privacy-and Security-by design Methodology Handbook. 2016*, p. 116.

1844 *See* Commission Nationale de l'Informatique et des Libertés, *Privacy Impact Assessment (PIA). Methodology*; Commission Nationale de l'Informatique et des Libertés, *Privacy Impact Assessment (PIA). Templates*.

1845 Comparing this phase with the steps of the DPIA illustrated in Chapter 2, Section 2.5.2, it should be noted that it embeds both the assessment of the

2. Analysing the proportionality and the necessity of data processing, and whether it protects data subjects' rights. The CNIL suggests explaining and justifying the choices related to all the data protection principles of Article 5 GDPR. These choices should be the best possible solutions. The assessment on the rights refers to the need to explain how the controller is expected to comply with Articles 12–22 and 28 of the GDPR. The CNIL provided a detailed template for assessing the protection of principles and rights[1846];
3. Assessing data protection risks that are associated with data security and ensuring they are properly addressed. This is the phase where the controller should identify threats, estimate and evaluate likelihood and severity, and find "planned controls", meaning safeguards related to the data being processed, at security and governance levels. The three main threats are illegitimate access to personal data, unwanted change and disappearance. In the first category of controls the authority includes: encryption, anonymisation, data partitioning, logical access control, logging, integrity monitoring, archiving, and paper document security. These may be considered examples of technical measures. Among the controls for ensuring security, the CNIL mentions workstation security, backups, network security, monitoring, hardware security, and protection against non-human sources of risk. At organisational levels the possible controls are management of rules, risk management, project and incident management, personnel management and supervision, and relations with third parties. These may be considered examples of organisational measures;
4. Documenting the process to monitor and re-iterate on it in a continuous improvement process. The CNIL's template divides the controls for checking the "unsatisfactory, planned improvement or acceptable" levels of compliance. The CNIL interestingly suggests preparing a visual representation of the planned controls and the risks through graphs. Any formal advice of the DPO should be documented.

Within the methodology and template, the CNIL released an extended and comprehensive knowledge base for conducting the DPIA[1847]. In this

---

need for the instruments and the systematic description of the processing envisaged for each processing operation and asset.

1846 Comparing this phase with the steps of the DPIA illustrated in Chapter 2, Section 2.5.2, it may be noted that analysis on need and proportionality should be performed in relation to the purpose of the processing.

1847 Commission Nationale de l'Informatique et des Libertés, *Privacy Impact Assessment (PIA). Knowledge basis*.

study the authority maps examples of types of risks and of outcomes of feared events, and proposes a method for estimating severity and likelihood, which are scaled from "negligible", "limited", "significant" to "maximum" levels.

After the classification of threats, the CNIL described the proposed "planned controls" mentioned above. As an example, encryption means making personal data unintelligible to anyone without access authorisation on the basis of symmetric or asymmetric techniques, and it shall follow specific measures[1848]. Encryption may be used for: equipment, databases, standalone files, email, and communication channels. Data partitioning is another control that reduces risks[1849]. The CNIL suggested separating the personal data necessary for each processing operation and creating different access rights to reduce the occurrence of data breaches. The large contribution of the CNIL is particularly valuable since it combines a methodology with know-how and state of the art measures, as ENISA usually does for security and data protection topics.

In 2019, the CNIL published open-source software for carrying out the DPIA called "PIA"[1850]. This tool is available for Windows, Linux and Mac OS operating systems, supports several languages, and has a user-friendly interface. PIA can be used as a legal and technical knowledge base for a data protection impact assessment on the basis of the GDPR and the CNIL framework. Since it provides a modular assessment, the data controller can easily customise this tool.

It should be underlined that despite the existence of methodologies and tools, every data controller should always specify and contextualise the assessment based on their context and business[1851].

Having defined a framework for the risk assessment, the following section describes techniques and standards to be taken into account during a DPbD approach.

1848 Commission Nationale de l'Informatique et des Libertés, *op. cit.*, pp. 14–17.
1849 Commission Nationale de l'Informatique et des Libertés, *op. cit.*, p. 18.
1850 *See* the official website at <www.cnil.fr/en>. Last accessed 06/10/2021.
1851 *See* the arguments in Sarrat and Brun, "DPIA: how to carry out one of the key principles of accountability".

5.5 Existing standards and PETs for EHR systems

This section summarises some existing standards and PETs that may be useful for the EHR implementation. It is out of the scope of this section to provide a taxonomy of the tools. The section presents recommended standards and a few PETs mentioned in the literature[1852].

As Hartzog noted, standards are crucial for implementing privacy and security since they guide compliance activities by providing useful and widely adopted specifications and solutions[1853]. Despite the fact that standards are usually not binding, they provide so-called best practices, and

1852 *See* J.A. Magnuson and Brian E. Dixon. *Public health informatics and information systems*. Springer, 2020. ISBN: 9783030412159; Josep Domingo-Ferrer and Alberto Blanco-Justicia. "Privacy-Preserving Technologies". In: *The Ethics of Cybersecurity*. Springer, Cham, 2020, pp. 279–297; AGID Agenzia per l'Italia Digitale. *Linee Guida per l'adozione di un ciclo di sviluppo di software sicuro*. Linee guida per lo sviluppo del software sicuro. Allegato 1, 2020; AGID Agenzia per l'Italia Digitale. *Linee Guida per la modellazione delle minacce e individuazione delle azioni di mitigazione conformi ai principi del Secure/Privacy by Design*. Linee guida per lo sviluppo del software sicuro. Allegato 4, 2020; Stefan Schulz, Robert Stegwee, and Catherine Chronaki. "Standards in healthcare data". In: *Fundamentals of Clinical Data Science*. Springer, Cham, 2019, pp. 19–36; Farina, *Il cloud computing in ambito sanitario tra security e privacy*; ENISA European Union Agency for Network & Information Security. *ICT security certification opportunities in the healthcare sector*. European Union Agency for Network and Information Security, 2018; Tamó-Larrieux, *Designing for privacy and its legal framework: data protection by design and default for the internet of things*; W. Ed Hammond. "Standards for Global health information systems". In: *Global Health Informatics*. Elsevier, 2017, pp. 94–108; European Union Agency for Network & Information Security, *Handbook on Security of Personal Data Processing*; Danezis et al., *Privacy and Data Protection by design – from policy to engineering*; J.A. Magnuson, Riki Merrick, and James T. Case. "Public Health Information Standards". In: *Public health informatics and information systems*. Springer, 2014, pp. 133–155. ISBN: 9780387227450; Sinha et al., *Electronic health record: standards, coding systems, frameworks, and infrastructures*; Hartley and Jones, *EHR implementation: A step-by-step guide for the medical practice*; Pierluigi Perri. *Privacy, diritto e sicurezza informatica*. Giuffrè Editore, 2007. ISBN: 8814137021, pp. 143–163; Cimino and Shortliffe, *Biomedical Informatics: Computer Applications in Health Care and Biomedicine*, pp. 265–311.

1853 Hartzog, *Privacy's blueprint: the battle to control the design of new technologies*, p. 164.

are useful for PbD, and DPbD[1854]. Nonetheless, it should be noted that standards are not free of charge[1855].

As regards ISO international standards on security and privacy, the following list identifies the key tools that provide guidance to data controllers and processors:

- ISO/Guide 73:2009(en) on risk management vocabulary, which was mentioned above, with the other ISO standards on this topic, which are ISO 31000:2018 and IEC 31010:2019 on risk management guidelines and risk assessment techniques respectively[1856];
- ISO/IEC 29100:2011 and ISO/IEC 29101:2018, which create a high-level privacy framework for processing in ICTs[1857]. ISO/IEC 29100 defines 11 privacy principles: "consent and choice; purpose legitimacy and specification; collection limitation; data minimisation; use, retention and disclosure limitation; accuracy and quality; openness, transparency and notice; individual participation and access; accountability; information security; and privacy compliance"[1858]. According to the standard, these principles should guide the design and development of ICTs;
- ISO/IEC 27001:2013, on information security management, which provides requirements at the organisational level, and ISO/IEC 27002:2013 on information security controls[1859]. ISO/IEC 27001 recommends creating an information security policy, organising roles and responsibili-

---

1854 *See* Kroener and Wright, "A strategy for operationalizing privacy by design", p. 362, which refers to PbD.

1855 *See* the comment in Tamó-Larrieux, *Designing for privacy and its legal framework: data protection by design and default for the internet of things*, p. 174; Magnuson, Merrick, and Case, "Public Health Information Standards", pp. 136–138.

1856 ISO, *ISO/Guide 73:2009(en) Risk management — Vocabulary*; ISO. *ISO 31000:2018 Risk management — Guidelines*. Tech. rep. ISO/TC 262, 2018; ISO. *IEC 31010:2019 Risk management — Risk assessment techniques*. Tech. rep. ISO/TC 262, 2019.

1857 ISO. *ISO/IEC 29100:2011 Information technology — Security techniques — Privacy framework*. Tech. rep. ISO/IEC, 2011; ISO. *ISO/IEC 29101:2018 Information technology — Security techniques — Privacy architecture framework*. Tech. rep. ISO/IEC, 2018. The amendment AMD 1:2018 was added to the first standard.

1858 Looking at these principles it may be argued that they followed the OECD Guidelines, the FIPs and the DPD's principles. *See* a discussion on the principles in Chapter 4, Section 4.2.

1859 ISO. *ISO/IEC 27001:2013(en) Information technology — Security techniques — Information security management systems — Requirements*. Tech. rep. ISO/IEC, 2013; ISO. *ISO/IEC 27002:2013 Information technology — Security techniques — Code of practice for information security controls*. Tech. rep. ISO/IEC, 2013.

ties, identifying security risks and planning actions for addressing these risks, and providing the resources for the security management system. The organisation should document the assessment, monitor security performance, and conduct internal audits;
- ISO/IEC 27035–1:2016 and ISO/IEC 27035–2:2016 on information security incident management, which present concepts for detecting, reporting, assessing, and responding to security incidents[1860];
- ISO/IEC 29134:2017, which provides guidance for privacy impact assessment[1861];
- ISO/IEC 27000:2018, on information security management systems and techniques, which explains the preservation of confidentiality, integrity, and availability[1862];
- ISO/IEC 27005:2018, on information security risk management, which is based on a recognised risk assessment approach[1863];
- ISO/IEC TS 19608:2018, which provides guidance for developing security and privacy functional requirements which are based on ISO/IEC 15408, an evaluation standard on IT security[1864];
- ISO/IEC 24760–1:2019 on identity management and privacy protection[1865]. This standard defined an identity management system as "mechanism comprising of policies, procedures, technology and other resources for maintaining identity information including associated metadata";

1860 ISO. *ISO/IEC 27035–1:2016 Information technology — Security techniques — Information security incident management — Part 1: Principles of incident management*. Tech. rep. ISO/IEC, 2016; ISO. *ISO/IEC 27035–2:2016 Information technology — Security techniques — Information security incident management — Part 2: Guidelines to plan and prepare for incident response*. Tech. rep. ISO/IEC, 2016. These standards are under review and will be replaced by ISO/IEC WD 27035–1.3 and ISO/IEC WD 27035–2.3.

1861 ISO. *ISO/IEC 29134:2017 Information technology — Security techniques — Guidelines for privacy impact assessment*. Tech. rep. ISO/IEC, 2017.

1862 ISO, *ISO/IEC 27001:2013(en) Information technology — Security techniques — Information security management systems — Requirements*.

1863 ISO. *ISO/IEC 27005:2018(en) Information technology — Security techniques — Information security risk management*. Tech. rep. ISO/IEC, 2018.

1864 ISO. *ISO/IEC TS 19608:2018 Guidance for developing security and privacy functional requirements based on ISO/IEC 15408*. Tech. rep. ISO/IEC, 2018; ISO. *ISO/IEC 15408–1:2009 Information technology — Security techniques — Evaluation criteria for IT security — Part 1: Introduction and general model*. Tech. rep. ISO/IEC, 2009.

1865 ISO. *ISO/IEC 24760–1:2019 IT Security and Privacy — A framework for identity management — Part 1: Terminology and concepts*. Tech. rep. ISO/IEC, 2019.

– ISO/IEC TR 27550:2019 on privacy engineering and system life cycle processes[1866];
– ISO/IEC 27701:2019, which extends ISO/IEC 27001 and ISO/IEC 27002 on privacy information management[1867];
– ISO/IEC 27007:2020 on information security management systems and auditing;
– ETSI TR 103 456, which is a European standard providing guidance on the NIS Directive on security of network and information systems[1868].

Additionally, as mentioned in Chapter 2, ISO/PC 317 is currently under development to provide the first international standard on privacy by design that will be applicable to any data processing involving consumer goods and services[1869].

During the implementation of the EHR system and its source systems two main areas of standards and PETs should at least be taken into account: interoperability and accessibility. Several ISO standards are specifically available for health informatics and EHR:

– As mentioned above, ISO standard 20514:2005(en) on the definition of EHR and EHR system[1870];
– ISO 18308:2011, which provides the requirements for an EHR architecture[1871]. This standard defines the structure of an EHR, which should store both clinical and administrative information, and should support authentication, data integrity, confidentiality, non-repudiation, and audit of accessed information[1872];

1866 ISO. *ISO/IEC TR 27550:2019 Information technology — Security techniques — Privacy engineering for system life cycle processes*. Tech. rep. ISO/IEC, 2019.

1867 ISO. *ISO/IEC 27701:2019 Security techniques — Extension to ISO/IEC 27001 and ISO/IEC 27002 for privacy information management — Requirements and guidelines*. Tech. rep. ISO/IEC, 2019. On this standard and the GDPR *see* Eric Lachaud. "ISO/IEC 27701 Standard: Threats and Opportunities for GDPR Certification". In: *Eur. Data Prot. L. Rev.* 6 (2 2020), pp. 194–210.

1868 ETSI. *ETSI TR 103 456 V1.1.1 (2017–10) Implementation of the Network and Information Security (NIS) Directive*. Tech. rep. ETSI/CYBER, 2017.

1869 *See* Chapter 2, Section 2.3, comment on line 13.

1870 *See* Chapter 3, Section 3.4.1 on ISO, *Health informatics — Electronic health record — Definition, scope and context. 20514:2005(en)*.

1871 ISO. *ISO 18308:2011 Health informatics — Requirements for an electronic health record architecture*. Tech. rep. ISO/TC 215, 2011.

1872 *See* the analysis by Sinha et al., *Electronic health record: standards, coding systems, frameworks, and infrastructures*, pp. 16–21. This article argues that the standard did not provide any details on these requirements.

- ISO 17090–1:2013 on digital certificate services, which will be replaced by ISO/DIS 17090–1[1873];
- ISO 22857:2013, which provides guidelines on data protection during trans-border flows of personal health data[1874];
- ISO 22600–1:2014 on privilege management and access control[1875];
- ISO/HL7 10781:2015 on EHR functional model, which provides the set of functional requirements, but is under review[1876];
- ISO 27799:2016 on information security of HITs, which is based on ISO/IEC 27002[1877];
- ISO 25237:2017 on pseudonymisation, that provides a basic methodology for techniques in the health care sector[1878];
- ISO 13606–1:2019, ISO 13606–2:2019, ISO 13606–3:2019, ISO 13606–4:2019, and ISO 13606–5:2019 on EHR communication architecture, its security, the privileges necessary to access the EHR data, and the

1873 ISO. *ISO 17090–1:2013 Health informatics — Public key infrastructure — Part 1: Overview of digital certificate services*. Tech. rep. ISO/TC 215, 2013.

1874 ISO. *ISO 22857:2013 Health informatics — Guidelines on data protection to facilitate trans-border flows of personal health data*. Tech. rep. ISO/TC 215, 2013.

1875 ISO. *ISO 22600–1:2014 Health informatics — Privilege management and access control — Part 1: Overview and policy management*. Tech. rep. ISO/TC 215, 2014; ISO. *ISO 22600–2:2014 Health informatics — Privilege management and access control — Part 2: Formal models*. Tech. rep. ISO/TC 215, 2014; ISO. *ISO 22600–3:2014 Health informatics — Privilege management and access control — Part 3: Implementations*. Tech. rep. ISO/TC 215, 2014.

1876 ISO. *ISO/HL7 10781:2015 Health Informatics — HL7 Electronic Health Records-System Functional Model, Release 2 (EHR FM)*. Tech. rep. ISO/TC 215, 2015.

1877 ISO. *ISO 27799:2016 Health informatics — Information security management in health using ISO/IEC 27002*. Tech. rep. ISO/TC 215, 2016.

1878 ISO. *ISO 25237:2017 Health informatics — Pseudonymization*. Tech. rep. ISO/TC 215, 2017.

1879 ISO. *ISO 13606–1:2019 Health informatics — Electronic health record communication — Part 1: Reference model*. Tech. rep. ISO/TC 215, 2019; ISO. *ISO 13606–2:2019 Health informatics — Electronic health record communication — Part 2: Archetype interchange specification*. Tech. rep. ISO/TC 215, 2019; ISO. *ISO 13606–3:2019 Health informatics — Electronic health record communication — Part 3: Reference archetypes and term lists*. Tech. rep. ISO/TC 215, 2019; ISO. *ISO 13606–4:2019 Health informatics — Electronic health record communication — Part 4: Security*. Tech. rep. ISO/TC 215, 2019; ISO. *ISO 13606–5:2019 Health informatics — Electronic health record communication — Part 5: Interface specification*. Tech. rep. ISO/TC 215, 2019.

interface specifications[1879]. ISO 13606 was originally designed by the European Committee for Standardization (CEN)[1880];

The standards on privacy management of personal health information in general, for privacy requirements of EHR systems, and audit trail of EHRs are currently under development in the ISO/TC 215 Technical Committee[1881].

Data format standards, vocabulary standards, and laboratory test and code standards are examples of categories of standards used for the EHR system and its source system[1882]. As an example, the Digital Imaging and Communications in Medicine (DICOM) standard provides the framework for communication and management of medical imaging information and related data[1883]. SNOMED CT standardised health terms that are globally used for EHRs, EMRs, PHRs systems and e-health technologies in general[1884].

Several different standards have been developed to achieve semantic interoperability[1885]. Among them, Health Level 7 (HL7) Group created

1880 European Union Agency for Network & Information Security, *ICT security certification opportunities in the healthcare sector*, p. 22, explains that the work of CEN aimed to create European standards that are harmonised with existing international standards.

1881 *See* ISO/AWI 22697 at <www.iso.org/standard/73697.html>. *See* ISO/AWI TS 14441 at <www.iso.org/ standard/80018.html>. *See* ISO/DIS 27789 at <www.iso.org/standard/75313.html>. Last accessed 06/10/2021.

1882 *See* the classification in Schulz, Stegwee, and Chronaki, "Standards in healthcare data"; Magnuson, Merrick, and Case, "Public Health Information Standards"; Sinha et al., *Electronic health record: standards, coding systems, frameworks, and infrastructures*; MITRE, *Electronic Health Records Overview*.

1883 *See* the official website at <www.dicomstandard.org/>. Last accessed 06/10/2021.

1884 SNOMED CT also has an ontological layer. *See* the official website at <www.snomed.org/>. Last accessed 06/10/2021.

1885 *See* Julien, "Electronic Health Records"; and Pulkit Mehndiratta, Shelly Sachdeva, and Sudhanshu Kulshrestha. "A model of privacy and security for electronic health records". In: *International Workshop on Databases in Networked Information Systems*. Springer. 2014, pp. 202–213, p. 204, which repors: "Health Level 7 (HL7), Clinical Document Architecture (CDA), CEN EN 13606 EHRcom, openEHR, Digital Imaging and Communications in Medicine Structured Reporting (DICOM SR), Web Access to DICOM Persistent Objects (ISO WADO), integrating the Healthcare Enterprise (IHE), Retrieve Information for Display (RID) and IHE Cross-Enterprise Document Sharing (XDS)".

the most widely implemented international standards for clinical-data interchange[1886].

HL7 defined standards and protocols for the structure of the data exchange both as messages and as documents[1887]. In particular, ISO/HL7 27931:2009 applies to the electronic data exchange in healthcare environments[1888], and ISO/HL7 21731:2014 provides the reference information model for the exchange[1889]. In the HL7 FHIR v. 4 protocols[1890], there are three privacy-related specifications: FHIR Security, FHIR Resource Consent and FHIR AuditEvent[1891]. These HL7 protocols have been included in the HIPAA's requirements[1892]. In addition, the HL7 FHIR framework released ontologies on health data that use the Web Ontology Language (OWL)[1893].

It is worth mentioning the openEHR project, which provides principles for creating an interoperable EHR systems software architecture that is based on a multilevel and single-source modelling framework[1894]. In

1886 *See* the information on this standard at the official website <www.hl7.org/>. Last accessed 06/10/2021. The history of the group was reported by Hammond, "Standards for Global health information systems"; and Cimino and Shortliffe, *Biomedical Informatics: Computer Applications in Health Care and Biomedicine*, pp. 300–302.

1887 ISO/HL7 27951:2009 Health informatics – Common terminology services, release 1 and ISO/HL7 27932:2009 Data Exchange Standards — HL7 Clinical Document Architecture, Release 2 are under review.

1888 ISO. *ISO/HL7 27931:2009 Data Exchange Standards — Health Level Seven Version 2.5 — An application protocol for electronic data exchange in healthcare environments*. Tech. rep. ISO/TC 215, 2009.

1889 ISO. *ISO/HL7 21731:2014 Health informatics — HL7 version 3 — Reference information model — Release* Tech. rep. ISO/TC 215, 2014.

1890 *See* <hl7.org/fhir/>. Last accessed 06/10/2021.

1891 A description of FHIR is provided by Hammond, "Standards for Global health information systems", pp. 103–104.

1892 *See* 45 C.F.R. § 170.215, § 170.299, § 170.315(d).

1893 *See* Athanasios Kiourtis et al. "Aggregating the syntactic and semantic similarity of healthcare data towards their transformation to HL7 FHIR through ontology matching". In: *International Journal of Medical Informatics* 132 (2019), p. 104002; Athanasios Kiourtis et al. "Structurally Mapping Healthcare Data to HL7 FHIR through Ontology Alignment". In: *Journal of Medical Systems* 43.3 (2019), pp. 62–75, which describes the knowledge base.

1894 Duarte Gonçalves-Ferreira et al. "OpenEHR and general data protection regulation: evaluation of principles and requirements". In: *JMIR medical informatics* 7.1 (2019), e9845. *See* also Kalra, Beale, and Heard, "The openEHR foundation"; Sinha et al., *Electronic health record: standards, coding systems, frameworks, and infrastructures*, pp. 163–174.

2003 the openEHR Foundation was established to openly publish EHR technical specifications, clinical models, open-source software, and several educational resources[1895]. The research created an information model that is separated from the content model, meaning that the logic structure of the EHR is defined in the first model while datasets are external. In 2019, this framework was tested for compliance with the GDPR. In particular, openEHR features have been matched with GDPR requirements. As an example, the legal requirement "period of storage limitation" is associated with the sentence "the system must allow the definition of deadlines for the processing of specific personal data, in order with the purpose of processing", and openEHR is scrutinised to assess whether it meets this requirement. The storage limitation principle, integrity, confidentiality, availability principles, interoperability, access rights and accountability are all matched in the openEHR project. Other requirements, however, have not yet been satisfied.

Cross-Enterprise Document Sharing (XDS) provides a standards-based specification for managing the sharing of documents, i.e. HIE, between different healthcare entities, ensuring interoperability[1896]. XDS can be used for national, regional or local EHR environments. This standard was developed by the US initiative called Integrating the Healthcare Enterprise (IHE), which has been active in promoting standards and solutions for healthcare communication service.

IHE also created a centralised access control system for the XDS environment: the Secure Retrieve (SeR) supplement[1897]. SeR functions with one authorisation decision manager. Therefore, it is not applicable where multiple data controllers use the EHR system. However, other IHE solutions may be useful in a complex EHR environment. The technical framework of IHE is even promoted by the European Commission[1898].

The IHE Basic Patient Privacy Consent (BPPC) provides a widely recognised mechanism to record patient's consent in a machine-readable

1895 *See* the mission of the Foundation at <www.openehr.org/about/vision_and_mission>. Last accessed 06/10/2021.

1896 *See* the information on XDS at <wiki.ihe.net/index.php/Cross-Enterprise_Document_Sharing>. Last accessed 06/10/2021.

1897 *See* the information on SeR at <wiki.ihe.net/index.php/Secure_Retrieve>. Last accessed 06/10/2021.

1898 *See* Commission Decision (EU) 2015/1302 of 28 July 2015 on the identification of 'Integrating the Healthcare Enterprise' profiles to reference in public procurement. O.J. L. 199, 29.7.2015.

form[1899]. Patient's consent is identified by a document with Extensible Markup Language (XML) that contains machine-readable indications. Despite the fact that IHE is a US-based developer, several policies available in the BPPC are applicable in the EU context. In fact, supportable policies are: "opt-in to clinical use" (which applies where consent is required by law), "specific document is marked as available in emergency situations" (which allows processing in a vital interest scenario), "additionally allow specific research project" (which applies to secondary use of personal data), "limit access to functional roles providers" and "limit access to structural roles" (which is fundamental in the EHR context). The BPPC is limited to a fixed list of policies. On the other hand, the Advanced Patient Privacy Consents (APPC) defines the structural representation necessary to capture, manage, and communicate patient's consent between systems and entities, independently of a set of policies. So, this solution seems more useful than BPPC for managing consent and access to EHR documentation.

As the EHR system involves several source systems, identity and access management are aspects where PETs are really useful. Several users may access the record with different duties, so techniques on secure accessibility are crucial. Access control is a typical security measure, which limits the risk that unauthorised entities might access the system[1900]. It has been pointed out that the most EHR systems incorporate access control mechanisms, but several different models may be adopted[1901].

1899 IHE International: Basic Patient Privacy Consent. IHE ITI TF Vol. 3 Section 5.0. This document was revised in June 2020 and is available at <www.ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_TF_Vol3. pdf>. *See* also the document of the European Commission on BPPC at <progressivestandards.org/standard/basic-patient-privacy-consents-ihe-bppc/>. Last accessed 06/10/2021.

1900 *See* e.g. in European Union Agency for Network & Information Security, *Handbook on Security of Personal Data Processing*; Commission Nationale de l'Informatique et des Libertés, *Privacy Impact Assessment (PIA). Knowledge basis*, pp. 24–27. *See* also security concepts in Agenzia per l'Italia Digitale, *Linee Guida per l'adozione di un ciclo di sviluppo di software sicuro*; Agenzia per l'Italia Digitale, *Linee Guida per la modellazione delle minacce e individuazione delle azioni di mitigazione conformi ai principi del Secure/Privacy by Design*; Perri, *Privacy, diritto e sicurezza informatica*, pp. 111–123.

1901 *See* Jorge Calvillo-Arbizu, Isabel Román-Martínez, and Laura M. Roa-Romero. "Standardized access control mechanisms for protecting ISO 13606-based electronic health record systems". In: *IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*. IEEE. 2014, pp. 539–542, which proposes a mechanism based on the eXtensible Access Control Markup Language (XACML).

The first solution for access control is following ISO 13606:2019 standard, which describes the identity management system. This is a high-level framework. Each entity should have specific attributes to be an identity and follow the identification and authentication process. The privacy-related capabilities of an identity management system are to[1902]:

- "implement mechanisms, including policies, processes; and technology, for minimal disclosure;
- authenticate entities that use identity information;
- minimize the ability to link identities;
- record and audit the use of identity information;
- protect against inadvertently generating risks to privacy, e.g. those posed by inadequately protecting identity information in logs and audit trails;
- implement policies for selective disclosure;
- implement policies to engage a human entity for explicit direction or consent, for activities related to their sensitive identity information".

So, within the implementation of an identity system, organisational policies and procedures should be set, and an audit control and record system should monitor the entity's activities.

Role-Based Access Control (RBAC) or Attribute-Based Access Control (ABAC) are two different privacy and security techniques that may be used in the EHR system. Within RBAC access to a system is granted on the basis of a defined user's role (e.g. professional category). The model implements several security principles, such as the separation of duties principle and is suited to an EHR context where the roles are limited and previously defined. In fact, a role has fixed privileges. ABAC, on the other hand, gives specific series of attributes and combines them with access policies. This model seems more suited to an EHR context where access rights are more granular and complex[1903]. However, the concrete solution to be implemented should be evaluated on a case-by-case basis.

Finally, the EHR system uses a network for information sharing and stores data in a repository. On the one hand several technologies and PETs can be used to secure the content of the communications, such as

---

1902 *See* ISO, *ISO 13606–1:2019 Health informatics — Electronic health record communication — Part 1: Reference model*.

1903 *See* on RBAC and ABAC Guasconi et al., *Reinforcing trust and security in the area of electronic communications and online services. Sketching the notion of "state-of-the-art" for SMEs in security of personal data processing*, pp. 18–19. *See* Danezis et al., *Privacy and Data Protection by design – from policy to engineering*, pp. 24–26.

encrypted channels or VPN[1904]; on the other hand, full disk encryption (FDE) techniques at the software or hardware level or file system-level encryption (FSE) are tools for protecting EHR data storage[1905].

This Chapter has described several tools for designing privacy and data protection in general and in the e-health context in particular. The next Chapter uses the theoretical and applied perspectives examined in these five chapters to provide a set of DPbD guidelines for the EHR system.

1904 *See* the description of several secure communication techniques in Danezis et al., *op. cit.*, pp. 27–31; Diffie and Landau, *Privacy on the line: The politics of wiretapping and encryption*, pp. 11–56. *See* also Commission Nationale de l'Informatique et des Libertés, *The CNIL's Guide on Security of personal data*, p. 13, which indicates both basic precautions and advanced techniques.

1905 *See* the analysis of encryption in Danezis et al., *Privacy and Data Protection by design – from policy to engineering*, pp. 40–42; Perri, *Privacy, diritto e sicurezza informatica*, pp. 125–142.