Contextual Integrity and Reasonable Expectations: A Privacy Paradigm

1st Amin Rabinia
School of Computing and Information Science
University of Maine
amin.rabinia@maine.edu

2nd Daniel Nathan
Department of Philosophy
Texas Tech University
daniel.nathan@ttu.edu

3rd Sepideh Ghanavati
School of Computing and Information Science
University of Maine
sepideh.ghanavati@maine.edu

***Abstract*—Due to the complex and ambiguous nature of privacy concerns, the transition between an abstract notion of privacy and practical aspects of privacy protection in IT systems becomes challenging. Even the existing privacy engineering approaches, such as privacy by design (PbD), do not mitigate this problem successfully. In this paper, we propose the notion of privacy paradigms as a comprehensive framework for developing privacy-preserving systems. Such paradigms aim at covering the entire life cycle of privacy engineering (i.e. requirements, design and development, and verification) and helping developers incorporating privacy into system development process. We also introduce a moral/legal theory of privacy and integrate it with practical guidelines of system development to instantiate a Privacy Paradigm (PriPa).**

***Index Terms*—Privacy by design, Privacy engineering, Requirements engineering, Contextual integrity**

I. Introduction

With the rapid expansion of data gathering and processing in information systems, data privacy concerns have become more alarming than ever. Privacy engineering is an emerging discipline to meet such concerns. The European Union Agency for Network and Information Security (ENISA) in its report on privacy [8] encourages further multidisciplinary investigations in privacy engineering to improve the understanding of system developers about privacy and data protection. This report, particularly, highlights the complications for concrete implementation of Privacy by Design (PbD) [19] [20] approach in the systems as demanded by the European General Data Protection Regulation (GDPR) [10]. In this paper, we integrate approaches from several disciplines such as philosophy of science, philosophy of law, ethics, and privacy engineering to develop a comprehensive IT privacy paradigm.

A comprehensive privacy paradigm is needed for various reasons: 1) Privacy is a vague notion for developers and its implementation as a high-level, non-functional requirement is challenging [9]; 2) Privacy regulations are not usually accompanied with a proper guideline for their interpretation and implementation. This results in compliance challenges for IT businesses, especially for small companies and startups that have limitations with respect to legal counseling; 3) IT businesses are asked to be proactive regarding their customers' privacy right. This demand, due to the sophistication of new information technologies and data processing techniques (e.g., Big Data, data mining, deep learning, etc.), has also become more important; 4) There is no comprehensive privacy engineering framework to systematically address user-centric privacy concerns [17]; 5) Lack of a comprehensive understanding of privacy and a coherent privacy engineering framework, leads to add-on privacy provisions and ad-hoc solutions.

The existing privacy approaches do not completely succeed in meeting these concerns. Privacy by Design (PbD) [19] [20] does not integrate a privacy theory with the design guidelines and has no compliance verification mechanism. Privacy Design Strategies [21] and Privacy Enhancing Techniques (PETs) [14] focus mainly on the technical solutions at the development level and do not address privacy concerns thoroughly throughout the Privacy Engineering Life Cycle (PELC) (i.e. requirements, design and development, and verification).

To mitigate the challenges raised by privacy concerns, we propose our Privacy Paradigm (PriPa) framework which includes the following contributions: 1) Developing a generic privacy paradigm that incorporates privacy theories with several system development strategies and techniques in order to overcome the theoretical and practical difficulties of privacy protection in information systems. 2) Outlining a prototype of PriPa, as an instance of privacy paradigms. 3) Theorizing privacy as consistency of the informational behaviors with the reasonable expectations. Any compliant information system, based on this paradigm, should behave in accordance with the accepted informational norms within a specific context, and therefore, must conform with the expectations of their users. This being said, the PriPa also entails a comprehensive system development guideline which covers the entire PELC. A privacy paradigm, such as the PriPa, helps system developers go beyond the scope of privacy regulations and proactively develop privacy-preserving systems.

The paper is organized as follows: In Section II, we review the methodology of our proposed research. In Section III, we describe the theoretical basis of the PriPa. Section IV outlines the practical contributions of the PriPa for developing privacy-preserving systems. Section V describes the plans for evaluation of our proposal while Sections VI provides an overview of the related work. Finally, in Section VII, we conclude our work and outline the future work.

II. Research Methodology

To develop our privacy paradigm, we take a qualitative research approach. We outline first, the ontology of a paradigm and then, the research methodology of our study.

A. The Concept of a Paradigm

In philosophy of science, a scientific paradigm [11], e.g. Newtonian physics, is a set of exemplars that defines what the scientific activity of the time should look like. Scientific paradigms entail *entities* such as theories, hypothesis, laws, methods, and problems. They also have three *functions*: 1) suggesting new problems; 2) suggesting approaches to solving those problems; 3) being the standard for measuring the success of a proposed solution [12]. Likewise, for performing these three functions in a systematic way, a privacy paradigm entails entities such as: privacy theories that define what privacy means, approaches or methods to identify privacy violations, and a set of privacy strategies/techniques to safeguard users' privacy in the design and development level.

B. Value Sensitive Design

The Value Sensitive Design (VSD) [5] is a design approach that targets a comprehensive implementation of human values in the system design processes. For example, it helps incorporating human values such as physical health, creativity, and emotional well-being in designing a workspace. The VSD has an integrative and iterative methodology, which involves three types of investigations: 1) Conceptual investigations, which deal with the recognition of direct and indirect stakeholders, their relationship with the system, the values at risk, and the trade-off among (moral/non-moral) values. 2) Empirical investigations, which are about measurements of success of a particular design in an actual human context. 3) Technical investigations, which deal with the possibility of technological infrastructure to support or threaten human values [5].

Since it is assumed that privacy is also one of the fundamental human values that needs to be thoroughly implemented in any systems, the VSD can be adopted as a meta-methodology for developing privacy paradigms. To develop the PriPa, we adopt VSD concepts. In the conceptual investigations, we define the theoretical framework of the PriPa (Section III). In the technical investigations, we develop practical guidelines of the PriPa, entailing PbD approach, for privacy-preserving system development (Section IV). Finally, in the empirical investigations, we perform the evaluation of the PriPa (discussed in Section VI). These investigations are non-sequentially repeated to develop the current prototype of the PriPa.

III. Conceptual Investigations (Theoretical Foundation)

What is the moral importance of privacy right? How might an answer to this question help us better protect data privacy? In this section, we propose an answer to the first question and in Section IV, we show how this answer helps creating privacy-preserving systems.

Our main goal is to, first, present a moral justification of privacy right, and then, link that to the practical aspects of privacy protection in IT. In fact, to protect individuals' data privacy, IT businesses cannot rely solely on privacy regulations. Beyond their legal commitments, they should also be morally devoted to protecting user's privacy. However, this moral devotion has never been well-justified for providing system developers with practical guidelines. We propose incorporating a moral theory into a useful, working, and guiding paradigm. In the following, we explain the moral/legal theories that underpin our privacy paradigm, the PriPa. It is important to know what a right to privacy means, and then, try to acknowledge this right and protect individuals' privacy. Without a comprehensive and consistent understanding of privacy, implementations of privacy provisions would be incomplete, ad-hoc, and case-by-case solutions. Here, we start by outlining some of the shortcomings of a prominent view of privacy as ”control over information”, and then, illustrate other alternatives, namely ”reasonable expectations” and ”contextual integrity”.

A. Privacy as Control over Information

One of the most dominant definitions of privacy in the legal and philosophical literature, as well as in IT domain, is in terms of control over information:”privacy is control over when and by whom the various parts of us can be sensed by others” [1]. Accordingly, since personal data are part of a person, a privacy right necessitates control by subjects over their information and restricting any type of unwanted access to it. This definition of privacy, however, suffers from an important shortcoming when it comes to information sharing. In fact, actual control over information is limited (or only applies) to the initial decision of whether and with whom to share. Past that decision point, by the initial act of sharing, one's control over the information ends. In other words, by sharing their information, individuals give up part of their privacy, qua control [2]. This is, however, exactly where a *right* to privacy should come into effect to protect individuals.

B. Privacy as Respect for Reasonable Expectations

When individuals share their information within a system, they still have a legitimate will or a reasonable expectation that their information is being treated fairly, legitimately, and trustfully within that system. A right to privacy, in this regards, is part of the universal human interests in having a realistic understanding of the world, in being able to predict the consequences of our actions, and in having our reasonable expectations fulfilled [3]. What we perceive as our surrounding environment forms our expectations about our world and gives us an understanding of our own actions within it. These expectations or understandings enable us to have mindful interactions within the environment. An environment consistent with our reasonable expectations is a necessity for a moral life. Conversely, an inconsistent environment violates our reasonable expectations and thus, disturbs our moral life. A privacy right is to ensure such consistency. For an information system, likewise, consistency of its informational behaviors with the reasonable expectations of its users is the hallmark of a proper privacy protection.

C. *Privacy as Maintaining Contextual Integrity*

When it comes to privacy, people have different types of expectations, which are not always the most reasonable. A complementary part to the notion of privacy as reasonable expectations is the theory of contextual integrity [4]. This theory determines what types of expectations are reasonable and should be considered in a privacy-preserving design. In Nissenbaum's view, two general types of informational norms play a role in the contexts: 1) norms of appropriateness, which determine what personal information is appropriate to be shared with in a particular context, and 2) norms of flow or distribution, which regulate transfer of information between parties within a context. To maintain contextual integrity within a system, both types of norms should be maintained [4].

Concerning the reasonableness of expectations, with respect to the context and its informational norms, system developers can decide what types of expectations are reasonably required to be considered in the system design. Users' expectations that are the natural outcome of the informational norms of a specific context should be assumed as the reasonable expectations. For example, based on norms of academia, visibility of a student's grades for the instructor is reasonably expected, but for other students is not.

To conclude this section, the PriPa integrates and adopts the two aforementioned privacy theories as its theoretical foundation. In a nutshell, the PriPa aims to ensure that information systems guarantee their contextual integrity and provide a consistent platform with users' reasonable expectations. [16] discusses the initial idea of this theory, as well. In the next section, we discuss the practical aspects of the PriPa.

IV. Technical Investigations (Technical Heuristics)

The PriPa, as a comprehensive privacy paradigm, needs to outline practical guidelines for implementation of its theoretical foundation. Recall that a paradigm does not suggest solutions to the given problems, but instead it proposes some heuristics (i.e. general guidelines or methods) for finding the legitimate solutions. In a coherent paradigm, such heuristics are in accordance with the theoretical foundation of the paradigm. In this section, we discuss the heuristics, drawn from PriPa's theoretical foundation, that help implementing PriPa's privacy theory throughout the privacy engineering phases of requirement, design and development, and verification. Fig. 1 shows PriPa's privacy engineering life cycle.

A. *Privacy Requirements*

The study [17] shows that there is a tangible gap between understanding or expectations of users and developers in terms of privacy concerns. It also concludes that most users are more concerned about their shared content, while developers pay more attention to the technical aspects of privacy protection. Regardless of the cause and nature of this difference, developers must always consider expectations of the users. However, as this study clarifies, there is no comprehensive privacy requirements framework to meet the user-centric privacy concerns [17]. Capturing users' expectations becomes more challenging in dynamic systems that include changing requirements, functions, contexts, technologies, number of users, etc [26]. As a result of this situation, and also due to lack of a comprehensive understanding of the concept of privacy, a weak privacy requirements leads to an inaccurate privacy design [18]. As discussed in [22], privacy requirements should go beyond the written regulations and make enough room for unpredicted interpretations. Existing approaches in legal requirements engineering are rarely equipped with such flexibility [23].

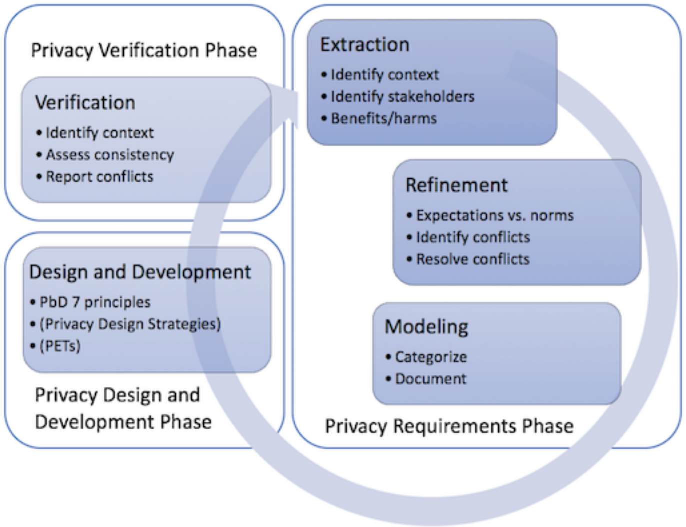

Fig. 1. PriPa's Privacy Engineering Life Cycle

To mitigate these problems, we define a heuristic for the PriPa and incorporate it with the methodology of VSD, to help developers identify potential privacy requirements of a system-to-be. This heuristic is based on the idea of contextual integrity and reasonable expectations. To use this heuristic, we need to divide privacy requirements phase into three stages of requirements extraction, refinement, and modeling.

a) Privacy Requirements Extraction: In this stage, we need to analyze the context in which the system-to-be is placed (e.g. healthcare, financial, academic, etc.) and identify its informational norms. For example, in a university context, advisors' access to students' enrollment is normal but access to students' dining plan is against the norms. To understand the specific reasonable expectations of the users, taking a user-centric perspective is required. For this, we should, for example, ask: ”what would be the reasonable expectations of a typical user about sharing data?”, ”what would users reasonably expect about the data access?”, etc. Having these investigational queries, the process of requirements extraction can be further articulated using the VSD guideline [5]:

1) **Identify the Context:** As we mentioned above, knowing

the context is the key for a privacy-preserving system, and thus, we should begin with identifying the context of privacy requirements.

2) **Identify Direct and Indirect Stakeholders:** In this step, individuals, groups, or parties that directly or indirectly interact with the system should be identified.
3) **Identify Benefits and Harms for Each Stakeholder Group:** In this step, it is crucial to consider reasonable expectations of the stakeholders (mainly users). For example, a student benefits from her advisor to access her enrollment system and, it is reasonably expected for her to share data with the advisor in the enrollment system. If a group of potential users are available for interview, extraction of expectations will be easier. Otherwise, we should take a user-centric perspective (using the investigational queries mentioned above) to accomplish this task.

b) Privacy Requirements Refinement: Having a solid knowledge of informational norms of a given context is necessary. Here, the extracted privacy requirements need to be further refined in order to be useful for the system design. For this purpose, the guideline continues as follows:

1) **Assess the Extracted Expectations vs. the Contextual Norms:** In this step, we assess the reasonableness of the extracted expectations. Given a set of informational norms within a context, we should confirm whether a user's expectation is reasonable or not. For example, advisor's access to student's information within the enrollment system, is a reasonable expectation. This expectation, however, is not reasonable in a different context, e.g. in the dining system.
2) **Identify Potential Conflicts:** There might be conflicts between different expectations (e.g. about student's academic performance and confidentiality), or between norms and expectations (for example, even if advisor can change the enrollment (it is expected) but it is against the norms to do so without informing the student). In the privacy requirement phase, not all conflicts are resolvable. The remaining conflicts should be flagged for further speculations in the design phase.
3) **Resolve the Conflicts if Possible:** If conflicts are resolvable in the requirements phase, they should be resolved. For example, conflicts between expectations about student's performance and confidentiality, can be resolved by prioritizing the expectations (the one that benefits the student more). Existing approaches such as Privacy Design Strategies [21] and PETs [14] can be appropriately used here.

c) Privacy Requirements Modeling: In this stage, the extracted and refined privacy requirements need to be properly documented. Categorizing requirements based on the type of informational behaviors (collection, process, share, etc.) and the stakeholder groups (users, third party, admin, etc.) can help designers in the design phase.

B. Privacy Design and Development

There are more studies [19] [20] [21] done on the design phase of privacy engineering than the requirements phase. However, these studies are still far from a theoretically coherent privacy framework. Privacy by Design (PbD), as the leading approach, has 7 foundational principles for incorporating privacy in the design process (Proactive not Reactive, Privacy as the Default Setting, Privacy Embedded into Design, Full Functionality, End-to-End Security, Visibility and Transparency, and Respect for the User) [20]. These principles do not completely follow a theoretical foundation, and although Cavoukian believes in privacy as control over information [19], she does not explicitly connect design principles of PbD with this privacy theory. Moreover, due to a weak mechanism to integrate privacy with the development process, PbD is notoriously hard to implement [8] [24] [15]. (It is not clear whether the new operationalized version of PbD [19] has been successful yet.) Consequently, PbD principles end up being merely a set of design advices.

Our privacy paradigm, the PriPa, cannot resolve all the difficulties that PbD deals with either. The PriPa, for example, does not suggest any development techniques. However, by integrating/interpreting PbD principles with the PriPa, we can achieve a better privacy-preserving design framework. This design framework clarifies what a proper privacy design looks like and leaves the options for fulfilling such a design open. In other words, it does not prescribe certain strategies or techniques for having a good design, but it provides standards for verifying a good design.

To this end, we begin with the theoretical foundation of PbD. As shown in Section III-A, the control theory of privacy is not a sufficient provision for protecting individuals' privacy. Instead, we can interpret PbD principles in the light of the theories of reasonable expectations and contextual integrity:

1) Proactive not Reactive: It is important to be proactive with respect to privacy protection, especially in dynamic systems that regulations cannot foresee the challenges and enforce the right provisions beforehand. A solid requirements modeling, using the heuristics outlined in Section IV-A, can help a proactive privacy design. Developers can identify what the privacy expectations are, and thus, design a system that meets such requirements.
2) Privacy as the Default Setting: Users expect privacy protection from the very beginning of their involvement in an enterprise. A trusted system does not need constant effort of users to protect their privacy. Default protection of privacy is the very basic expectation of users from a trusted system. Thinking ahead, in privacy requirements phase (Section IV-A), immensely helps a privacy by default approach.
3) Privacy Embedded into Design: When developers internalize PriPa's theory, i.e. consistency of the system with the users' expectations, there would be nothing to be added afterward in the system to satisfy this criterion. Every bit of system design should be in accordance with

this hallmark and be privacy-preserving.

4) Full Functionality: As we discussed in Section IV-A, the conflicts between expectations should be prioritized. There should be no dilemma between privacy and system's functionalities. Individuals usually consent to give up part of their privacy (e.g. share their academic performance) but not their privacy right (e.g. their right to confidentiality). Thus, protection of privacy as a right should not be seen as a conflict with system's functionalities.
5) End-to-End Security: Security is one of the fundamental informational norms and users' expectations in any system. Without a strong security, there will be no ground for protection of privacy.
6) Visibility and Transparency: Another fundamental informational norm is transparency. No one wants to interact with an obscure system. Transparency makes it possible for users to verify whether the system is still consistent with their expectations or not. It is also the key for informed consent. For the continuation of informed consent, users should be notified for any change in the system. Unnoticed changes of the system can disrupt users' expectations and thus, violate their privacy.
7) Respect for the User: Privacy, based on the PriPa, is nothing but respect for the users. Respecting users means respecting their choices, as autonomous persons, and their reasonable expectations, as the precondition of their choices. Any privacy-preserving system needs to behave in a consistent manner with their user's reasonable expectations and a respecting design approach should provide such consistency.

The PbD principles lack a unifying theoretical insight. The interpretations based on PriPa's insight, provided above, make PbD principles to be understood in light of a supporting theoretical foundation. The ENISA report [8] states that there are conceptual difficulties for safeguarding privacy when developers are not familiar with privacy as a different set of requirements than the usual functional requirements. This can be even harder when design approaches are not accompanied with concrete guidelines. The rationale behind a privacy paradigm is to internalize privacy concepts for developers, so that they better deal with privacy concerns, especially during the design process. It is usually easier to answer how-questions when we already know the answers to what- and why-questions. The PriPa aims at answering the questions of what privacy is and why it is important. Knowing these answers helps developers figure out the answer to the question of how to implement privacy provisions. Although, developers are free to choose their own design architectures, or techniques, the PriPa's main objective is to provide means to verify if it is an acceptable design (i.e. if it is consistent with the norms and expectations).

C. *Privacy Verification*

The GDPR (Art. 41) requires data controllers to periodically conduct data privacy compliance reviews. Even though running such reviews demands different types of investigations (e.g. legal, technical, policy, etc.), PriPa's privacy theory helps identifying potential or actual privacy violations. For this, we repeat some steps of the privacy requirements phase (i.e. Section IV-A):

1) **Identify the Context:** We should identify the context of the system, including its elements such as stakeholders, types of data, data practices (e.g. collection, process, share), and processing purposes.
2) **Assess the Consistency of the System:** In this step, it is required to assess the consistency of the implemented system versus the PriPa's hallmark of privacy. This entails confirming whether the system can maintain users' expectations and conform with the norms. We should specifically certify that no data practice in the system violates the reasonable expectations of typical users or the contextual norms. For example, we should search for any unexpected (i.e. without permission) data access of other stakeholders; or any extension of data use beyond the user's expectations (i.e. the legitimate and notified purposes).
3) **Report the Identified Conflicts:** Any identified conflict alongside its causes should be reported to the appropriate group (requirements engineers or system designers). We repeat the entire PELC (Fig. 1) until all the potential conflicts are identified and resolved.

This being said, the PriPa's privacy verification process does not substitute conducting a thorough legal compliance review with legal experts. Applying PriPa's guideline is simply a precursory measure to design a privacy-preserving system under a certain understanding of privacy and does not necessarily satisfy all the detailed requirements set by regulations.

V. Preliminary Evaluation and Discussion

As part of VSD's empirical investigation, we need to repeatedly evaluate our privacy paradigm. A preliminary conceptual evaluation of the PriPa has been done on few examples of privacy violation incident [1] to assess the strength of its theoretical foundation. However, a thorough empirical evaluation of the entire paradigm is needed as it is still a proposed approach. To show how adequate the PriPa is in practice, we plan to perform separate evaluations on three phases of the PriPa by asking non-author developers to use the PriPa during few experimental IT projects. The objective of such evaluations is to determine whether the PriPa facilitates recognition and resolution of privacy problems during the system development life cycle. In the requirements phase, for example, we will study how much the proposed heuristic (in IV-A) is helpful for the developers to identify privacy expectations of users and refine them as privacy requirements. In the design phase, we will measure how helpful the PriPa is for developers to avoid privacy violations. Finally, in the verification phase, we analyze the success of developers in identifying privacy violations using the PriPa. In our work, we follow design

[1] Temporary link: https://tinyurl.com/y3hdtcx7

science approaches [25] and thus, after each set of evaluation, we plan to refine our paradigm to make sure it resolves the shortcomings of the proposed approach.

VI. Related Work

There are various methodologies, with different concerns and levels of abstraction, in privacy engineering. Some approaches such as PbD [19] [20] are motivated by legal guidelines (e.g. Fair Information Practice Principles) and address privacy concerns from a regulatory perspective. However, as discussed earlier, PbD has shortcomings in meeting technical needs of system design and in compliance verification process.

On the technical side, Privacy Design Strategies [21] and Privacy Enhancing Technologies [14] provide practical provisions for developing privacy-preserving systems. However, they lack a broad perspective to cover all aspects of the privacy engineering life cycle, especially in requirements phase.

To operationalize the GDPR, [13] suggests a 6-steps systemic approach that helps developers identifying solution requirements. This approach, however, is limited to requirements phase and is not based on a specific theory of privacy.

Research [6] proposes a security engineering method for incorporating privacy requirements, as organizational goals, into system design. In [7], a framework for implementing security and privacy by design is offered in the domain of the Internet of Things. These approaches, however, due to their narrow definition of privacy and scope of application fail to fully cover the nuances of privacy in system development.

VII. Conclusion and Future Work

In this paper, we defined a new approach for addressing privacy concerns in information system development, which involves formulating privacy engineering frameworks in terms of privacy paradigms, consist of theories, methods, and techniques. Such paradigms serve as comprehensive guidelines, covering the entire privacy engineering life cycle, for developing privacy-preserving systems. As an example, we proposed a prototype of our privacy paradigm, the PriPa.

Based on the PriPa, a privacy-preserving system upholds its users’ reasonable expectations and maintains the contextual norms. To help implementing such systems, the PriPa defines a heuristic guideline to include privacy concept early on and employ it throughout the system development process.

Since this work is still a proposed approach, in future, we plan to perform a comprehensive evaluation of the PriPa through case studies and experimental implementations by including developers and end-users. We also intend to complete the prototype of the PriPa with finer-grained technical guidelines at the design and development level, with Privacy Design Strategies.

References

[1] R. B. Parker, ”A definition of privacy,” in Privacy, Routledge, 2017, pp. 83-104.

[2] W. A. Parent, ”Privacy, Morality, and the Law,” Philosophy and Public Affairs, vol. 12, no. 4, pp. 269-288, 1983.

[3] D. O. Nathan, ”Just looking: Voyeurism and the grounds of privacy,” Public Affairs Quarterly, vol. 4, no. 4, pp. 365-386, 1990.

[4] H. Nissenbaum, ”Privacy as contextual integrity,” Wash. L. Rev., vol. 79, p. 119, 2004.

[5] B. Friedman, P. H. Kahn, A. Borning, and A. Huldtgren, ”Value Sensitive Design and Information Systems,” in Early engagement and new technologies: Opening up the laboratory, N. Doorn et al Eds. Springer, 2013, pp. 55-95.

[6] C. Kalloniatis, E. Kavakli, and S. Gritzalis, ”Addressing privacy requirements in system design: the PriS method,” Requirements Eng, vol. 13, no. 3, pp. 241-255, Sep. 2008.

[7] N. Foukia, D. Billard, and E. Solana, ”PISCES: A framework for privacy by design in IoT,” in 2016 14th Annual Conference on Privacy, Security and Trust (PST), 2016, pp. 706-713.

[8] ”Privacy and Data Protection by Design ENISA.” [Online]. Available: https://www.enisa.europa.eu/publications/privacy-and-data-protection-by-design. [Accessed: 17-Apr-2019].

[9] S. Schiffner et al., ”Towards a roadmap for privacy technologies and the General Data Protection Regulation: A transatlantic initiative,” in Annual Privacy Forum, 2018, pp. 24-42.

[10] ”General Data Protection Regulation (GDPR)- Final text neatly arranged,” General Data Protection Regulation (GDPR). [Online]. Available: https://gdpr-info.eu/. [Accessed: 17-Apr-2019].

[11] T. S. Kuhn, The structure of scientific revolutions. University of Chicago press, 2012.

[12] A. Bird, ”Thomas Kuhn,” Aug. 2004.

[13] V. Ayala-Rivera and L. Pasquale, ”The Grace Period Has Ended: An Approach to Operationalize GDPR Requirements,” in IEEE 26th Int. Requirements Engineering Conference (RE), 2018, pp. 136-146.

[14] J. J. Borking, ”Why Adopting Privacy Enhancing Technologies (PETs) Takes so Much Time,” in Computers, Privacy and Data Protection: an Element of Choice, S. Gutwirth et al Eds. Dordrecht: Springer Netherlands, 2011, pp. 309-341.

[15] I. S. Rubinstein and N. Good, ”Privacy by Design: A Counterfactual Analysis of Google and Facebook Privacy Incidents,” Berkeley Tech. L.J., vol. 28, p.1333, 2013.

[16] A. Rabinia, ”Privacy in Information Technology,” Thesis, 2018. Available: https://hdl.handle.net/2346/82723. [Accessed: 17-Apr-2019].

[17] S. Sheth, G. Kaiser, and W. Maalej, ”Us and them: a study of privacy requirements across North America, Asia, and Europe,” in Proceedings of the 36th Int. Conference on Software Eng., 2014, pp. 859-870.

[18] M. Gharib, P. Giorgini, and J. Mylopoulos, ”Ontologies for privacy requirements engineering: A systematic literature review,” arXiv preprint arXiv:1611.10097, 2016.

[19] A. Cavoukian, Operationalizing privacy by design: A guide to implementing strong privacy practices. Information and Privacy Commissioner of Ontario. 2012.

[20] A. Cavoukian, ”Privacy by design: The 7 foundational principles,” Information and Privacy Commissioner of Ontario, Canada, vol. 5, 2009.

[21] J.-H. Hoepman, ”Privacy design strategies,” in IFIP International Information Security Conference, 2014, pp. 446-459.

[22] A. Rabinia and S. Ghanavati, ”FOL-Based Approach for Improving Legal-GRL Modeling Framework: A Case for Requirements Engineering of Legal Regulations of Social Media,” in 25th Int. Requirements Eng. Conference Workshops (REW), 2017, pp. 213-218.

[23] G. Boella, L. Humphreys, R. Muthuri, P. Rossi, and L. van der Torre, ”A critical analysis of legal requirements engineering from the perspective of legal practice,” in 7th Int. Workshop on Requirements Eng. and Law (RELAW), 2014, pp. 14-21.

[24] B.-J. Koops and R. Leenes, ”Privacy regulation cannot be hardcoded. A critical comment on the ’privacy by design’provision in data-protection law,” Int. Review of Law, Computers & Tech., vol. 28, no. 2, pp. 159-171, 2014.

[25] R. J. Wieringa, What Is Design Science?, in Design Science Methodology for Information Systems and Software Engineering, R. J. Wieringa, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 3-11.

[26] P. Anthonysamy, A. Rashid, and R. Chitchyan, ”Privacy requirements: present & future,” in 39th Int. Conference on Software Eng.: Software Engineering in Society Track, 2017, pp. 1322.