Towards A Knowledge Graph-based Exploratory Search for Privacy Engineering

Guntur Budi Herwanto[1,2,3], Fajar J. Ekaputra[4,3], Florina Piroi[3] and Marta Sabou[4]

[1]Faculty of Computer Science, University of Vienna
[2]Department of Computer Science and Electronics, Universitas Gadjah Mada
[3]Institute of Information Systems Engineering, Faculty of Informatics, TU Wien
[4]Institute of Data, Process, and Knowledge Management, WU Wien

Abstract

The concept of privacy-by-design has gained considerable attention in the wake of legal requirements that software systems should fulfill. The interconnected knowledge on privacy concepts, legal requirements, and software development artifacts required to implement this concept can be a major burden for organizations. To address this challenge, we propose a knowledge graph-based Exploratory Search system to help organizations complete privacy engineering tasks. We identify major requirements of such systems and develop an Exploratory Search prototype built on privacy engineering knowledge that integrates different information sources. While still preliminary, this work serves as a foundation for future research on integrating privacy knowledge into software development and demonstrates the potential of Exploratory Search Systems to support the cognitive process in privacy tasks such as privacy threat identification.

Keywords

knowledge graph, exploratory search, privacy engineering, privacy requirement

1. Introduction

The concept of "privacy by design" has come to the spotlight following the implementation of the General Data Protection Regulation (GDPR[1]). This concept mandates that software developers embed privacy protection directly into their applications from the start, rather than as an afterthought [1]. However, compliance with these privacy measures can present substantial challenges for software developers [2]. *Privacy Engineering* has recently emerged as a research area focusing on tackling these challenges [3].

Given the complexity of these various areas of knowledge, Privacy Engineering stakeholders, such as privacy engineers and software developers are often intimidated by the amount of knowledge required to follow privacy engineering principles [4]. They have to rely on various heterogeneous sources to ensure that their work covers the most relevant aspects of privacy requirements. Therefore, it is essential to support them with methods and tools for (i) integration of data from heterogeneous sources, and (ii) intuitive exploration of knowledge.

VOILA! 2023: 8th International Workshop on Visualization and Interaction for Ontologies and Linked Data, Athens, Greece, Co-located with ISWC 2023.

gunturbudi@ugm.ac.id (G. B. Herwanto); fajar.ekaputra@wu.ac.at (F. J. Ekaputra); florina.piroi@tuwien.ac.at (F. Piroi); marta.sabou@wu.ac.at (M. Sabou)

CEUR Workshop Proceedings (CEUR-WS.org)

[1]General Data Protection Regulation (GDPR), https://gdpr-info.eu/

The heterogeneous sources range from privacy frameworks, software artifacts, threat intelligence, privacy design patterns to region-specific regulations such as the GDPR and CCPA [5]. Utilizing semantic web technologies, especially ontologies, addresses this by offering a unified knowledge representation, seamlessly interlinking related concepts across diverse datasets. This interconnected framework ensures holistic privacy engineering, facilitating intuitive queries across software design [6], threats, and legal requirements, and promoting scalability as privacy landscapes evolve. Such technologies transform the intimidating breadth of information into a comprehensible and actionable resource for stakeholders.

While ontology-based Exploratory Search strategies have been successfully implemented in domains such as software engineering [6, 7] and the general domain [8], their application in the field of Privacy Engineering remains underexplored. Such strategies hold significant potential for assisting privacy engineers and software developers. Specifically, they can aid in the identification, connection, and modeling of privacy threats and mitigation strategies, processes which are currently conducted manually [9]. The exploration of this approach within Privacy Engineering could fill an essential gap in the literature and practice.

In this paper, we propose the adaptation of an existing Exploratory Search system method [6] and tool [7] equipped with necessary visualization for the privacy engineering context. For this purpose, we utilized an early version of an ontology that we developed, tailored for the privacy engineering context that covers most of the early stages of privacy-aware software development. This knowledge can be encapsulated and made reusable by such a system, streamlining the integration and implementation of privacy-related elements in design activities and, ultimately, ensuring compliance [5].

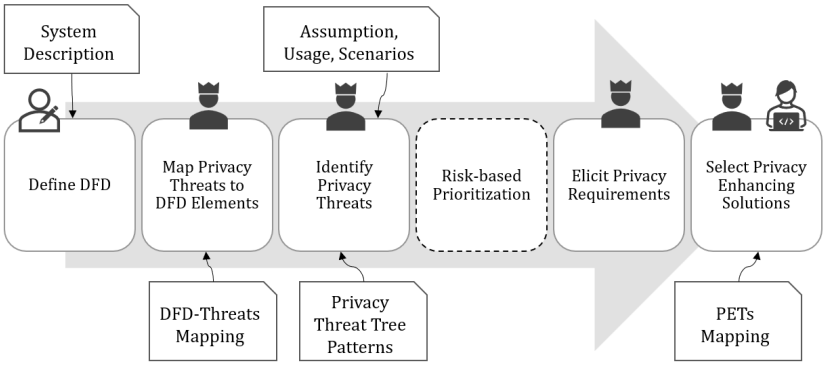

Figure 1: LINDDUN Privacy Threat Analysis Procedure [10]. The SAs shapes the DFD. The PEs elicits privacy threats and requirements, leading to the selection and development of privacy-enhancing solutions together with the SDs.

2. Context and Background

Figure 1 depicts a privacy threat modeling process called LINDDUN [10] consisting of six steps: (i) Data Flow Diagram (DFD) definition, (ii) Mapping privacy threats to DFD elements, (iii) Identification of misuse/problematic case scenarios, (iv) Risk-based prioritization of privacy issues, (v) Elicitation of privacy requirements, and (vi) Selection of privacy-enhancing solutions.

We selected LINDDUN as our model for privacy requirements engineering due to its extensive application and widespread acceptance [11]. Threat modeling involves the collaboration of numerous stakeholders. We outline three primary participants in the privacy engineering procedure: (i) *Software Architects (SA)*, who focuses on designing the general architecture of the software, e.g., DFD definition (cf. Step 1 in Figure 1), (ii) *Privacy Engineer (PE)*, who is responsible for identify privacy requirements and propose mitigation strategies (i.e., Step 2-6 in Figure 1, and (iii) *Software Developer (SD)*, who is responsible to implement technical measures according to the chosen mitigation strategies. In this paper, we focus on the role of PE and SD, whose scenarios and requirements will be described next.

Privacy Engineer (PE). A typical privacy engineering process often begins with a PE's review of DFD, which is created by the SA. For a more comprehensive understanding of the DFD, the PE may revisit the initial system requirements. After that, the PE can start the threat analysis, beginning with a particular element within the DFD. This analysis will involve various queries, such as "*Could this data flow pose a potential threat to the system? What type of threats could emerge? If such threats exist, how can they be mitigated?*".

To answer these questions, the PE explores the threat tree, which helps in identifying potential threats. Within the LINDDUN framework, seven unique threat types exist, each having its own hierarchical structure of threat trees [10]. Following the identification of potential threats, mitigation strategies can be suggested. These could involve using Privacy Enhancing Technologies (PETs) as per the existing mapping or creating a new, innovative solution beyond the current mapping.

Software Developer (SD). SDs are primarily responsible for implementing the technical measures outlined by privacy engineers. Nevertheless, SD technical expertise can contribute to a threat modeling process that includes identifying threats and selecting technical measures.

2.1. Exploratory Search Requirements

Based on our analysis on the typical scenarios of Privacy Engineering described the previous section, we identified a set of requirements to support PEs and SDs in their role in Privacy Engineering as the following:

Multiperspective Exploration. Multiple stakeholders involved in the privacy engineering process [10]. SAs provides DFDs as part of the system architecture. PEs ensure that privacy risks are identified and develop requirements for mitigating the risk. They can also recommend mitigation. The SDs will ensure that all suggested requirements and mitigations are incorporated into the system. Allowing all knowledge to be explored in one location by multiple stakeholders ensures consistency and traceability.

Data Flow Visualization. An important aspect of privacy threat modeling is understanding the data flow within a system, as this helps PEs identify potential threats. The DFD itself can be represented as a triple consisting of two elements (either external entity, process, or data store) associated with the data flow [12]. These elements also form the basis for threat identification [10]. Incorporating this data flow knowledge into the knowledge graph and visualizing it as a DFD within the exploratory search would be beneficial to the threat modeling process and provide traceability to the identified threat.

Threat Tree Visualization. Threat trees, also known as attack trees, are graphical repre-

sentations of threats or attacks against a system that are organized hierarchically. They show different ways a system can be exploited by breaking down higher-level threats into smaller, more specific threats. Therefore, a hierarchical browsing capability for threat trees is important. In addition, a text-based search would be beneficial to enable PEs to search for specific threats within the threat trees.

3. Exploratory Search Systems for Privacy Engineering

Based on the identified requirements from Section 2.1, we developed an initial Exploratory Search System for Privacy Engineering following the STAR approach [6]. We first set up the ontology and populate the knowledge graphs from existing privacy engineering datasets [10, 13, 12, 14]. Afterwards, we adapt our prior Exploratory Search systems framework [7] for this scenario.

3.1. Knowledge Graphs Construction

Ontology for Privacy Engineering. The ontology builds on concepts derived from privacy engineering methods [10, 15, 16]. The ontology aims to connect software development artifacts [13] with privacy knowledge [5]. The software development artifacts include the requirements, which may be in the form of user stories in agile requirements, and the DFD that represents them. Meanwhile, privacy knowledge might include the knowledge base about the personal data involved, privacy threats, privacy goals, legal requirements, and privacy mitigation strategies in the form of privacy design patterns. The ontology can be accessed on our GitHub page[2].

Ontology Population Privacy engineering tools such as ProPAN [15] or PrivacyStory [14] are examples of privacy engineering tools that would support the privacy engineering processes shown in Figure 1. These tools stored their results in various data models and formats and therefore will need to be transformed into an integrated format. In our prototype development, we use the knowledge generated by PrivacyStory[3] and transform them into the the previously described ontology. In the future, we plan to integrate more resources, e.g., the threat knowledge base and mitigation tools can be fed from the known threat knowledge base, and their mitigation can be mapped based on the known ontology [9]. Legal concepts such as GDPR can also be added to the knowledge base.

The details on the development and evaluation of the ontology and the population process from privacy engineering knowledge bases are beyond the scope of this paper.

3.2. Exploratory Search System Prototype

The implementation of the Exploratory Search System (ESS) is constructed upon the foundation laid by Haller et al. [7]. Their ESS is oriented toward the manufacturing sector. To accommodate our need for privacy knowledge within software development, we leveraged its capabilities and configured it to meet the needs of privacy engineering tasks. The implementation is accessible

[2]https://github.com/gunturbudi/ptm-ontology
[3]https://doi.org/10.5281/zenodo.8198322

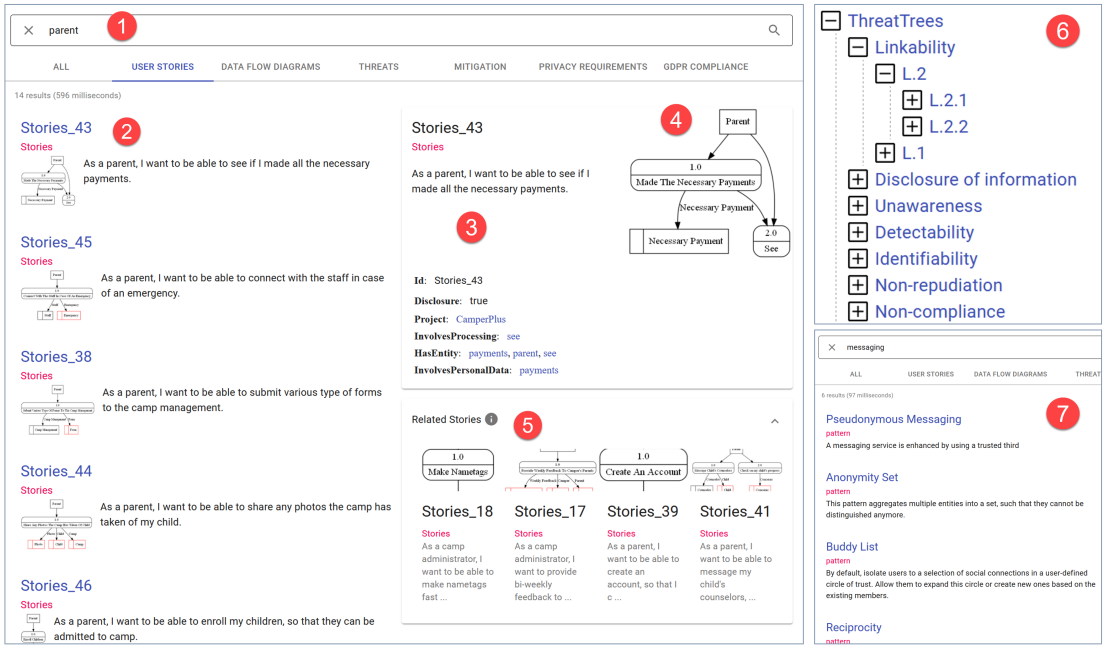

Figure 2: Screen captures of the exploratory search systems, comprising (1) the entry area for text searches, (2) the outcome of a full-text search of the keyword "parent", (3) an information box displaying descriptions and characteristics, (4) a data flow diagram's depiction, (5) A list of user stories related to the present selected user story, (6) a display of a hierarchical threat tree (can be accessed from the Hierarchy menu), and (7) the results of a mitigation search of the keyword "messaging".

online[4], and a screenshot of the ESS can be seen in Figure 2. In the ESS for Privacy Engineering, we includes the identified exploration components from Section 2.1, which will be briefly explained in the following.

Multiperspective Exploration. Allowing all knowledge to be explored in one location by multiple stakeholders ensures consistency and traceability. The ESS system can effectively use the multiperspective exploration feature for this purpose [7]. These perspectives are intentionally designed to provide valuable insight tailored to different stakeholders without overwhelming them with information.

Figure 3 shows the perspective of PEs and SDs. The privacy engineering process begins with the selection of user stories. They can select based on specific actors or stakeholders. After choosing a particular story, the privacy engineer can view other user stories that involve the same actor. These stories come with a DFD produced by privacy engineering tools [12], providing a visual representation of how data moves through different processing. This point marks the start of exploration for threat modeling purposes.

SDs are able to see what technical measures were chosen by the PEs, while also being able to trace back why those measures were chosen and for what user stories.

Data Flow Visualization. We included DFD visualizations for every user story to enhance

[4]http://privacy.semantics.id

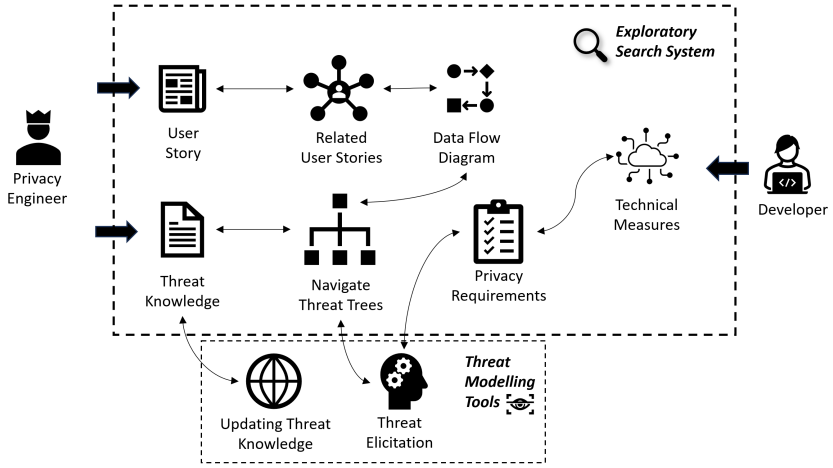

Figure 3: Exploratory scenario of PEs and SDs. PEs initiate exploration from User Story or Threat Knowledge, whereas SDs begin from technical measures. ESS, intended for exploration, draws its updates from external systems, as illustrated with threat modeling tools.

comprehension from the user's viewpoint [12]. In addition, we've converted the details within the DFD and its triple connections into a format that is traceable and searchable in text, making it easier to locate using text-based searches. PrivacyStory also allows us to present a single DFD that encompasses multiple user stories [14]. In the future, we are planning to represent DFD in a machine-readable manner for a better user experience.

Threat Tree Visualization. LINDDUN provides a catalog of privacy threat trees grouped into seven categories of threats [10]. We've converted this catalog into a knowledge graph, which is displayed hierarchically in the ESS. The tree-like hierarchical representation can be accessed through the hierarchy menu, which is separate from the main text search.

The ESS also allows users to start with the text-based search to efficiently identify related topics. The underlying text search uses indexing based on the triple store. In our system, we use Lucene scoring, which uses a combination of the vector space model (VSM) of information retrieval and the Boolean model to determine how relevant a particular document is to a user's query. Moreover, the ESS is capable of showing threats related to the currently displayed ones, thereby aiding users in navigating the threat knowledge base. Owing to its exploratory design, users have the flexibility to traverse back and forth within the interconnected knowledge base, thus expanding their understanding of related details and context. In future works, we aim to enhance this threat knowledge base by incorporating other privacy threat modeling approaches or knowledge [17].

4. Conclusion

Knowledge about privacy-by-design principles is currently scattered across different domains and locations, making it difficult for organizations to access and understand it due to the significant cognitive effort required. In this paper, we present an ontology-based search system

to facilitate organizations' access to privacy-related knowledge in relation to their own setting. The search system requires the organization to adapt its knowledge to the ontology and include it in the ontology that meets the standards. This system enables various stakeholders, including privacy engineers and developers, to visualize and understand data flow diagrams and threat trees through its networked knowledge base. While still in its early stages, this research lays the groundwork for future studies aimed at more effectively incorporating privacy knowledge in software development processes. It also highlights the potential of the Exploratory Search systems in reducing the cognitive demands associated with acquiring privacy knowledge.

In the future, we plan to evaluate how effectively privacy engineers or developers can use our exploratory search system to retrieve specific privacy knowledge and how they evaluate their user experience. We also plan to conduct comparative studies comparing the performance of our search system in facilitating the privacy engineering process with performance without the tools. Feedback from these evaluations will help us improve the ontology structure, user interface design, and overall user interaction flow.

Acknowledgments: This work is supported by the European Union's Horizon 2020 research project OntoTrans under Grant Agreement No 862136.

References

[1] V. Ayala-Rivera, L. Pasquale, The grace period has ended: An approach to operationalize gdpr requirements, in: 2018 IEEE 26th International Requirements Engineering Conference (RE), IEEE, 2018, pp. 136–146.

[2] A. Senarath, N. A. Arachchilage, Why developers cannot embed privacy into software systems? an empirical investigation, in: Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018, 2018, pp. 211–216.

[3] S. Gürses, J. M. Del Alamo, Privacy engineering: Shaping an emerging field of research and practice, IEEE Security & Privacy 14 (2016) 40–46.

[4] A. Senarath, M. Grobler, N. A. G. Arachchilage, Will they use it or not? investigating software developers' intention to follow privacy engineering methodologies, ACM Transactions on Privacy and Security (TOPS) 22 (2019) 1–30.

[5] J. C. Caiza, Y.-S. Martín, D. S. Guamán, J. M. Del Alamo, J. C. Yelmo, Reusable elements for the systematic design of privacy-friendly information systems: A mapping study, IEEE Access 7 (2019) 66512–66535.

[6] M. Sabou, F. J. Ekaputra, T. Ionescu, J. Musil, D. Schall, K. Haller, A. Friedl, S. Biffl, Exploring enterprise knowledge graphs: A use case in software engineering, in: The Semantic Web: 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, Proceedings 15, Springer, 2018, pp. 560–575.

[7] K. Haller, F. J. Ekaputra, M. Sabou, F. Piroi, Enabling exploratory search on manufacturing knowledge graphs, in: Proceedings of the 7th International Workshop on the Visualization and Interaction for Ontologies and Linked Data, volume 3253, 2022, pp. 16–28.

[8] A. G. Nuzzolese, V. Presutti, A. Gangemi, S. Peroni, P. Ciancarini, Aemoo: Linked data exploration based on knowledge patterns, Semantic Web 8 (2017) 87–112.

[9] A. Al-Momani, K. Wuyts, L. Sion, F. Kargl, W. Joosen, B. Erb, C. Bösch, Land of the lost: privacy patterns' forgotten properties: enhancing selection-support for privacy patterns, in: Proceedings of the 36th Annual ACM Symposium on Applied Computing, 2021, pp. 1217–1225.
[10] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, W. Joosen, A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements, Requirements Engineering 16 (2011) 3–32.
[11] K. Wuyts, L. Sion, W. Joosen, Linddun go: A lightweight approach to privacy threat modeling, in: 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), IEEE, 2020, pp. 302–309.
[12] G. B. Herwanto, G. Quirchmayr, A. M. Tjoa, From user stories to data flow diagrams for privacy awareness: A research preview, in: International Working Conference on Requirements Engineering: Foundation for Software Quality, Springer, 2022, pp. 148–155.
[13] G. B. Herwanto, G. Quirchmayr, A. M. Tjoa, A named entity recognition based approach for privacy requirements engineering, in: 2021 IEEE 29th International Requirements Engineering Conference Workshops (REW), IEEE, 2021, pp. 406–411.
[14] G. B. Herwanto, G. Quirchmayr, A. M. Tjoa, Privacystory: Tool support for extracting privacy requirements from user stories, in: 2022 IEEE 30th International Requirements Engineering Conference (RE), IEEE, 2022, pp. 264–265.
[15] R. Meis, Problem-Based Privacy Analysis (ProPAn): A Computer-aided Privacy Requirements Engineering Method, Universitaet Duisburg-Essen (Germany), 2018.
[16] M. Gharib, P. Giorgini, J. Mylopoulos, Copri v. 2—a core ontology for privacy requirements, Data & Knowledge Engineering 133 (2021) 101888.
[17] C. Bloom, Privacy threat modeling, USENIX Association, Santa Clara, CA (2022).