Literature Survey on Technologies for Developing Privacy-Aware Software

Atsuo Hazeyama, Tokyo Gakugei University
Hironori Washizaki, Waseda University
Nobukazu Yoshioka, National Institute of Informatics
Haruhiko Kaiya, Kanagawa University
Takao Okubo, Institute of Information SECurity
Case-based management system for secure and privacy-aware software development

Contents

1. Introduction (background and goal)
2. Privacy, privacy property and meta-model
3. Report on the current status of the body of knowledge for privacy-aware software development
4. Discussion
5. Summary
Introduction

• With the spread of cloud computing and highly functional mobile devices, anyone can use a variety of services
  ⇒ Risks regarding the leakage of private information have increased in number
• The need to tackle privacy issues in the early stage of the software development cycle called Privacy by Design (PbD) is recognized
• A body of knowledge for privacy-aware software development is required [Balebako+2014]
• Final goal: to create a knowledge base for technologies for privacy-aware software development
• Goal of this paper: to report on the current status of the body of knowledge required for privacy-aware software development based on a literature survey and to discuss a future direction

Contents

1. Introduction (background and goal)
2. Privacy, privacy property and meta-model
3. Report on the current status of the body of knowledge for privacy-aware software development
4. Discussion
5. Summary
Privacy

• Westin’s definition [Westin 1967]: an individual’s right “to control, edit, manage, and delete information about them[elves] and decide when, how, and to what extent information is communicated to others”

• Spiekermann and Cranor [Spiekermann+2009]: two dimensions for building privacy-friendly technologies and information systems

  ✓ Users can exercise immediate control over access to themselves and their personal data. This corresponds to the definition by Westin.
  ✓ Engineers are responsible for minimizing future privacy risks by protecting data.

### Privacy property [Deng+2011]

<table>
<thead>
<tr>
<th>Privacy property</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unlinkability</td>
<td>To hide the link between two or more actions, identities, and pieces of information</td>
</tr>
<tr>
<td>Anonymity</td>
<td>To hide the link between an identity and an action or a piece of information</td>
</tr>
<tr>
<td>Pseudonymity</td>
<td>To use pseudonyms as identifiers</td>
</tr>
<tr>
<td>Plausible deniability</td>
<td>The ability to deny having performed an action that other parties can neither confirm nor contradict</td>
</tr>
<tr>
<td>Undetectability and unobservability</td>
<td>To hide the user’s activities.</td>
</tr>
<tr>
<td>Confidentiality</td>
<td>To hide the data content or controlled release of data content.</td>
</tr>
<tr>
<td>Content awareness</td>
<td>To make sure that users are aware of their personal data and that only the minimum necessary information should be sought and used to allow for the performance</td>
</tr>
<tr>
<td>Policy and consent compliance</td>
<td>To require the whole system to inform the data subject about the system’s privacy policy or allow the data subject to specify consents</td>
</tr>
</tbody>
</table>

Meta-model for security and privacy for cloud services

Procedure used in literature collection

1. Google Scholar is used for the search engine and “Privacy pattern,” “Privacy requirement,” “Privacy Enhanced Technology (PET),” and “Privacy by Design (PbD)” are given as keywords for the search.

2. Read relevant papers for the topics from the search results and some other papers raised as references.
Collected literature
Knowledge items and studies that belong to them in this literature survey

<table>
<thead>
<tr>
<th>Knowledge item (class)</th>
<th>Explanation</th>
<th>Literature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Methodology/process</td>
<td>It is a technique, including concept and/or notation, and procedure.</td>
<td>[Deng+2011], [Kalloniatis+2008], [Sabo+2013], [Radics+2013], [Kung2014], [Martin+2014]</td>
</tr>
<tr>
<td>Principle</td>
<td>It is prescriptive knowledge [Barnum+2005]. Guideline is also prescriptive one. Principle is more abstract and philosophical than the guideline.</td>
<td>[Cavoukian2009], [OECD2013], [Hoepman2014]</td>
</tr>
<tr>
<td>Guideline</td>
<td>It is prescriptive knowledge [Barnum+2005]. Principle is also prescriptive one. Guideline is more concrete than the principle.</td>
<td>[Cavoukian+2014], [Spiekermann+2009]</td>
</tr>
<tr>
<td>Pattern</td>
<td>Pattern is, “design solutions to common privacy problems - a way to translate ‘privacy-by-design’ into practical advice for software engineering.”</td>
<td>[Deng+2011], [Kalloniatis+2008], [Privacypattern], [Hoepman2014], [Xuan+2014], [Hafiz2006], [Graf+2010], [Romanosky+2006], [Lobato+2009], [Chung+2004]</td>
</tr>
</tbody>
</table>

Contents

1. Introduction (background and goal)
2. Privacy, privacy property and meta-model
3. Report on the current status of the body of knowledge for privacy-aware software development
4. Discussion
5. Summary

Overview of studies that propose relationships among multiple classes
Methodology/Process

LINDDUN\textsuperscript{[Deng+2011]}

- Methodology to elicit privacy requirements
  \(\Leftarrow\) cf. threat modeling for security based on STRIDE (Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, and Elevation of privilege)

- Procedure
  1. Define a data flow diagram (DFD)
  2. Map privacy threats (LINDDUN) to DFD elements
  3. Identify misuse case scenarios
  4. Conduct Risk-based prioritization
  5. Elicit privacy requirements
  6. Select privacy enhancing solutions

- Seventeen threat tree pattern catalog

Methodology/Process

**LINDDUN [Deng+2011]**

Privacy threats and privacy properties

*Table 2* In the LINDDUN methodology, privacy properties and the corresponding privacy threat are categorized as hard privacy and soft privacy

<table>
<thead>
<tr>
<th>Privacy properties</th>
<th>Privacy threats</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unlinkability</td>
<td>Linkability</td>
</tr>
<tr>
<td>Anonymity &amp; Pseudonymity</td>
<td>Identifiability</td>
</tr>
<tr>
<td>Plausible deniability</td>
<td>Non-repudiation</td>
</tr>
<tr>
<td>Undetectability &amp; Unobservability</td>
<td>Detectability</td>
</tr>
<tr>
<td>Confidentiality</td>
<td>Disclosure of information</td>
</tr>
<tr>
<td>SOFT</td>
<td></td>
</tr>
<tr>
<td>Content awareness</td>
<td>content Unawareness</td>
</tr>
<tr>
<td>Policy and consent compliance</td>
<td>policy and consent Noncompliance</td>
</tr>
</tbody>
</table>
**Methodology/Process**

LINDDUN \cite{Deng2011}

**Table 4** Mapping LINDDUN components (privacy threats) to DFD element types (E-Entity, DF-Data flow, DS-Data store, P-Process)

<table>
<thead>
<tr>
<th>Threat categories</th>
<th>E</th>
<th>DF</th>
<th>DS</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linkability</td>
<td>×</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Identifiability</td>
<td></td>
<td>×</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Non-repudiation</td>
<td></td>
<td></td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Detectability</td>
<td></td>
<td></td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Information Disclosure</td>
<td></td>
<td></td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>content Unawareness</td>
<td></td>
<td></td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>policy/consent Noncompliance</td>
<td>×</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Threat tree pattern developed in LINDDUN (attack pattern)

- Linkability of entity
- Linkability of data flow
- Linkability of data store
- Identifiability of entity
- Identifiability of data flow
- Identifiability of data store
- Identifiability of process
- Non-repudiation of data flow
- Non-repudiation of data store
- Non-repudiation of process
- Detectability of data flow
- Detectability of data store
- Detectability of process
- Information disclosure of data flow, data store, and process
- Content unawareness of entity
- Consent and policy noncompliance of the system (data flow, process, and data store)
Methodology/Process

LIINDDU[N](Deng+2011)

<table>
<thead>
<tr>
<th>LINDDUN threats</th>
<th>Elementary privacy objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linkability of ((E,E))</td>
<td>Unlinkability of ((E,E))</td>
</tr>
<tr>
<td>Linkability of ((DF,DF))</td>
<td>Unlinkability of ((DF,DF))</td>
</tr>
<tr>
<td>Linkability of ((DS,DS))</td>
<td>Unlinkability of ((DS,DS))</td>
</tr>
<tr>
<td>Linkability of ((P,P))</td>
<td>Unlinkability of ((P,P))</td>
</tr>
<tr>
<td>Identifiability of ((E,E))</td>
<td>Anonymity / pseudonymity of ((E,E))</td>
</tr>
<tr>
<td>Identifiability of ((E,DF))</td>
<td>Anonymity / pseudonymity of ((E,DF))</td>
</tr>
<tr>
<td>Identifiability of ((E,DS))</td>
<td>Anonymity / pseudonymity of ((E,DS))</td>
</tr>
<tr>
<td>Identifiability of ((E,P))</td>
<td>Anonymity / pseudonymity of ((E,P))</td>
</tr>
<tr>
<td>Non-repudiation of ((E,DF))</td>
<td>Plausible denial of ((E,DF))</td>
</tr>
<tr>
<td>Non-repudiation of ((E,DS))</td>
<td>Plausible denial of ((E,DS))</td>
</tr>
<tr>
<td>Non-repudiation of ((E,P))</td>
<td>Plausible denial of ((E,P))</td>
</tr>
<tr>
<td>Detectability of (DF)</td>
<td>Undetectability of (DF)</td>
</tr>
<tr>
<td>Detectability of (DS)</td>
<td>Undetectability of (DS)</td>
</tr>
<tr>
<td>Detectability of (P)</td>
<td>Undetectability of (P)</td>
</tr>
<tr>
<td>Information Disclosure of (DF)</td>
<td>Confidentiality of (DF)</td>
</tr>
<tr>
<td>Information Disclosure of (DS)</td>
<td>Confidentiality of (DS)</td>
</tr>
<tr>
<td>Information Disclosure of (P)</td>
<td>Confidentiality of (P)</td>
</tr>
<tr>
<td>Content Unawareness of (E)</td>
<td>Content awareness of (E)</td>
</tr>
<tr>
<td>Policy and consent Noncompliance of the system</td>
<td>Policy and consent compliance of the system</td>
</tr>
</tbody>
</table>
PEAR architecture design methodology for PbD

(1) Extract functional privacy requirements and quality attribute privacy requirements

• Functional privacy requirements: the what part (i.e. what the system does) (purpose limitation, purpose specification, collection, use, disclosure and retention limit)

• Quality attribute privacy requirements: the how part (i.e. how the system does it). minimization, enforcement, accountability and modifiability

(2) Consider tactics for PbD

• Minimization: focuses on the minimization of disclosed information.
  ✓ anonymous credentials tactics which allow users to authenticate themselves without revealing their identity.
  ✓ tactics which make sure that computing is restricted to a given perimeter (ex. In the user client device)

• Enforcement
  ✓ data protection policies enforcement (ex. access rights)
  ✓ processing protection

• Accountability
  ✓ logging of relevant events (ex. the removal of data according to retention policies could be logged)
  ✓ protection of logging (the logging data must be unforgeable).

• Modifiability: cope with evolution needs (policy change, crypto change, and protection change)

(3) Associate tactics with architecture patterns.

ex.

- User data confinement pattern: to collect and process personal data in a location that is physically controlled by the user.
- Hippocratic management pattern: to enforce the collecting and processing of data in a predefined confinement area.

→ The resulting architecture associates data with metadata describing protection policies.
- Isolation pattern: provides isolation between independent applications.

→ To prevent the risk for other applications to accidentally or maliciously access computing resources and data associated with another application.

Relationships between tactics and architecture patterns are weak.

(4) Analyze and evaluate privacy architecture

ATAM (Architecture Tradeoff Analysis Method) and CBAM (Cost Benefit Analysis Method) can be used for privacy architecture analysis and evaluation.

Case study: electronic tolling pricing system

- Application requirements: record information about the vehicle route, ensure accuracy of the recorded information, bill the driver/vehicle based on the recorded route information, and keep the information for invoice verification.
- Non-functional requirements for privacy protection
  - User control of data
    - Keep the driven route private from road toll collectors, and keep route specific information only during the invoice litigation period.
      → a minimized data set consisting of the customer identification (required to associate a vehicle with a customer account and billing information), the proof data of the driven route, and the associated cost.
  - Policies associated with manipulated data
    - Detailed data concerning the vehicle route are kept and processed only inside the On-Board Equipment
    - Proof about the route driven is stored by the OBE only for a limited time that is needed for invoice verification
    - Data sent outside the OBE only contain the customer ID and the cost for the distance
    - Aggregated data is signed by the OBE to prevent tampering

(C) A. Hazeyama, H. Washizaki, N. Yoshioka, H. Kaiya and T. Okubo
### Seven principles proposed by Cavoukian [Cavoukian2009]

<table>
<thead>
<tr>
<th>Name</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proactive Not Reactive; Preemptive Not Remedial</td>
<td>PbD anticipates and prevents privacy invasive events before they happen.</td>
</tr>
<tr>
<td>Privacy as the Default Setting</td>
<td>Privacy data protection is built into the system, by default.</td>
</tr>
<tr>
<td>Privacy Embedded into Design</td>
<td>PbD is embedded into the design and architecture of IT systems and business practices. The result is that privacy becomes an essential component of the core functionality being delivered</td>
</tr>
<tr>
<td>Full Functionality - Positive-Sum, not Zero-Sum</td>
<td>PbD seeks to accommodate all legitimate interests and objectives in a positive-sum “win-win” manner, not through a dated, zero-sum approach.</td>
</tr>
<tr>
<td>End-to-End Security Full Life-Cycle Protection</td>
<td>PbD ensures that all data are securely retained, and then securely destroyed at the end of the process, in a timely fashion.</td>
</tr>
<tr>
<td>Visibility and Transparency - Keep it Open</td>
<td>Component parts of PbD and operations remain visible and transparent, to both users and providers alike.</td>
</tr>
<tr>
<td>Respect for User Privacy – Keep it User-Centric</td>
<td>Keep it user-centric.</td>
</tr>
</tbody>
</table>

Each principle in [Cavoukian+2014] was refined into sub-principles, and associated with guidelines.

Ex: Correspondence of sub-principle to guideline regarding “Privacy Embedded into Design”

<table>
<thead>
<tr>
<th>Sub-principle</th>
<th>Guideline</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1—Holistic and Integrative: Privacy commitments must be embedded in holistic and integrative ways.</td>
<td>SHALL use the OASIS PbD-SE Privacy Use Template or the more comprehensive OASIS PMRM methodology or equivalent for identifying and documenting privacy requirements</td>
</tr>
<tr>
<td>3.2—Systematic and Auditable: A systematic approach should be adopted that relies upon accepted standards and process frameworks, and is amenable to external review.</td>
<td>SHALL contain description of its business model showing traceability of personal data flows for any data collected through new software services under development.</td>
</tr>
<tr>
<td>3.3—Review and Assess: Detailed privacy impact and risk assessments should be used as a basis for design decisions.</td>
<td>SHALL include identification of privacy design principles</td>
</tr>
<tr>
<td>3.4—Human-Proof: The privacy risks should be demonstrably minimized and not increase through operation, misconfiguration, or error.</td>
<td>SHALL contain a privacy architecture</td>
</tr>
<tr>
<td></td>
<td>SHALL describe privacy UI/UX design</td>
</tr>
<tr>
<td></td>
<td>SHALL define privacy and security metrics</td>
</tr>
<tr>
<td></td>
<td>SHALL include human sign-offs/privacy checklists for software engineering artifacts</td>
</tr>
<tr>
<td></td>
<td>SHALL include privacy review reports (either in reviewed documents or in separate report)</td>
</tr>
</tbody>
</table>
Principle

Privacy design strategy [Hoepman2014]

• Hoepman advocated the concept of “privacy design strategy” and showed the relationship between elements of the concept and patterns

✓ Minimize: The amount of personal data that is processed should be restricted to the minimal amount possible
✓ Hide: Any personal data and their interrelationships should be hidden from plain view
✓ Separate: Personal data should be processed in a distributed fashion, in separate compartments whenever possible.
✓ Aggregate: Personal data should be processed at the highest level of aggregation and with the least possible detail in which it is (still) useful.
✓ Inform: Data subjects should be adequately informed whenever personal data is processed (transparency).
✓ Control: Data subjects should be provided agency over the processing of their personal data.
✓ Enforce: A privacy policy compatible with legal requirements should be in place and should be enforced.
✓ Demonstrate: A data controller should be able to demonstrate compliance with the privacy policy and any applicable legal requirements.

Documentation guideline for software engineers

- Represent privacy concerns in a UML diagram
  - Introduce containers to represent privacy properties (principles) in a use case diagram, place them between actors and use cases, and connect relationships between them
  - Represent objects and/or principles in an activity diagram

Guideline

Documentation guideline for software engineers (cont.)

- Represent privacy concerns in a UML diagram
  ✓ Represent to check privacy in a sequence diagram
Overview

• Requirement pattern: [Xuan+2014]
• Threat tree pattern: [Deng+2011]
• Design pattern: [Hafiz2006]
• UI pattern: [Graf+2010], [Romanosky+2006]
• Privacy policy pattern: [Lobato+2009]
• Patterns for ubiquitous computing: [Chung+2004]
• Process pattern: [Kalloniatis+2008]
• Privacy pattern catalog: [Hoepman2014], [privacypattern]

Numerous pattern catalogs are in isolation
Privacy pattern

Threat tree pattern [Deng+2011]

- Threat tree pattern developed in LINDDUN (attack pattern)

  ✓ Linkability of entity
  ✓ Linkability of data flow
  ✓ Linkability of data store
  ✓ Identifiability of entity
  ✓ Identifiability of data flow
  ✓ Identifiability of data store
  ✓ Identifiability of process
  ✓ Non-repudiation of data flow
  ✓ Non-repudiation of data store
  ✓ Non-repudiation of process
  ✓ Detectability of data flow
  ✓ Detectability of data store
  ✓ Detectability of process
  ✓ Information disclosure of data flow, data store, and process
  ✓ Consent unawareness of entity
  ✓ Consent and policy noncompliance of the system (data flow, process, and data store)

Fig. 4 Threat tree for linkability of an entity

Process pattern [Kalloniatis+2008]

Process patterns presented in PriS method

- Authentication pattern
- Authorization pattern
- Identification pattern
- Data protection pattern
- Anonymity and pseudonymity pattern
- Unlinkability pattern
- Unobservability pattern

<table>
<thead>
<tr>
<th>Privacy design strategy</th>
<th>Pattern</th>
</tr>
</thead>
</table>
| Minimize                | • Select before you select  
                            • Anonymisation and use pseudonyms |
| Hide                    | • Encryption  
                            • Mix networks  
                            • Attribute based credentials  
                            • Anonymisation and use pseudonyms  
                            • Differential privacy |
| Separate                |         |
| Aggregate               | • Aggregation over time  
                            • Dynamic location granularity  
                            • k-anonymity  
                            • l-diversity |
| Inform                  | • Platform for privacy preferences (P3P)  
                            • Data breach notification  
                            • Pattern collection by [Graf] |
| Control                 |         |
| Enforce                 | • Access control  
                            • Sticky policies |
| Demonstrate             | • Privacy management systems  
                            • Use of logging and auditing |

Contents

1. Introduction (background and goal)
2. Privacy, privacy property and meta-model
3. Report on the current status of the body of knowledge for privacy-aware software development
4. Discussion
5. Summary
Discussions

• Requirements from the viewpoint of developers who use such knowledge: seamless support from a privacy requirement analysis to the design phase, using the methodology or process, as well as the principles, guidelines, and patterns

⇨ to develop relationships among various types of knowledge

• Current status

✓ LINDDUN: relationship between methodology and threat tree pattern
✓ PEAR: relationship between methodology and pattern
✓ Pris method: relationship between methodology and process pattern
✓ Study by Hoepman: relationship between principle and pattern

Numerous pattern catalogs are in isolation

Contents

1. Introduction (background and goal)
2. Privacy, privacy property and meta-model
3. Report on the current status of the body of knowledge for privacy-aware software development
4. Discussion
5. Summary
Summary

• This paper introduced studies addressing knowledge regarding the development of privacy-aware software
  ✓ used a meta-model for security and privacy by Washizaki et al. to categorize each piece of knowledge
  ✓ collected different studies on methodology/process, as well as principle, guideline, and pattern.

  Contribution: one step toward the building of a body of knowledge for privacy-aware software development

• Future work
  ✓ Improvement of the coverage of literature collection
  ✓ Building a body of knowledge for privacy-aware software development
    ⇒ Enhancement of building of relationships among pieces of knowledge
      (numerous patterns are in isolation)
Acknowledgment

This study was partially supported by JSPS KAKENHI Grants Number 15H02686. This study was also supported by IISF SSR Forum 2015 and 2016. The authors would like to thank the reviewers for their valuable review comments.
Literature Survey on Technologies for Developing Privacy-Aware Software

Atsuo Hazeyama, Tokyo Gakugei University
Hironori Washizaki, Waseda University
Nobukazu Yoshioka, National Institute of Informatics
Haruhiko Kaiya, Kanagawa University
Takao Okubo, Institute of Information Security