Privacy-enabled Management of Customer Data

Günter Karjoth, Matthias Schunter, and Michael Waidner
IBM Research, Zurich Research Laboratory,
Säumerstrasse 4, 8803 Rüschlikon, Switzerland
{gka,mts,wmi}@zurich.ibm.com

Abstract

Customers are required to trust selected enterprises to protect their personal information. The large amounts of personal data that these enterprises store constitutes a risk. Accidental disclosure of personal data or misuse by employees often leads to bad publicity and can result in substantial liabilities. The first step towards proper privacy management is to inform the customers how their data is used and to whom it may be disclosed. This can be declared in a privacy notice that can be formalized using the W3C Platform for Privacy Preferences (P3P). Unfortunately, P3P does not provide means to enforce these privacy promises throughout and across multiple enterprises. This article describes technology for privacy-enabled management and exchange of customer data. Using a comprehensive privacy-specific access control language, we can express restrictions on the access to personal data, possibly shared between multiple enterprises. We separate the enterprise-specific deployment policy from the privacy policy that covers the complete life cycle of collected data. In addition, we introduce a viable separation of duty between the three "administrators" of a privacy system: The privacy officer designs and deploys privacy policies, the security officer designs access control policies, and the customers can give consent while selecting opt-in and opt-out choices.

1 Introduction

There is no viable technology than enables consumers to enforce proper use of their personal information throughout an enterprise[1]. As a consequence, customers are required to trust an enterprise once they disclose their personal data.

Many enterprises are aware of this risk and of the market share they might loose if they violate the trust of their customers. As a consequence enterprises publish privacy statements that promise fair information practices. Written in natural language or formalized using P3P [6], they merely constitute privacy promises and are not necessarily backed up by technological means.

In this article, we describe technology that enables an enterprise to enforce the privacy promises made to its customers. The goal is to enable the management of an enterprise (including the Chief Privacy Officer) to prevent misuse or inappropriate disclosure of personal data by regular employees. It solves different aspects of privacy enforcement:

Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

[1]Note that this goal of protecting data (provided by a customer) against its holder (enterprise) would be similar to the problem of Digital Rights Enforcement.

- Enterprises store a variety of personally identifiable information (PII or *personal data* for short). Enterprises may not know what privacy statement applies to what piece of data.
- Enterprises may not know the consent a customer has given nor the legal regulations that apply to a specific customer record.
- Enterprises exchange customer data. The privacy statement fixes usage restrictions at the point of collection. It is unclear how to communicate these restrictions once personal data is disclosed to another enterprise.

Whenever an enterprise collects, stores, or processes personal information, our concepts can be used to ensure that the data flows and usage practices of an enterprise comply with the privacy statement of that enterprise.[2] We cover the following areas:

- *Formalized Privacy Policies:* The policy language "Enterprise Privacy Authorization Language (EPAL)" [2] enables an enterprise to formalize a privacy policy into a machine-readable language that can be enforced automatically.
- *Formalized Policy Options:* An EPAL privacy policy can identify opt-in as well as opt-out choices or options that depend on the collected data (e.g., whether the given data pertains to a child). These options enable a company to use a limited number of policies while still providing freedom of choice to its customers.
- *Policy Enforcement:* Given collected personal data and its policy, the policy needs to be enforced. Policy enforcement covers several cooperating enterprises if personal data is exchanged among them. The core technology is a scheme for privacy-enabling access control that allows only actions that are authorized by the applicable privacy policy. Besides granting or denying access, privacy obligations have to be enforced as well (such as "we delete collected data if consent is not given within 15 days").
- *Compliance Audit:* A challenge of a privacy audit is that privacy policies are intertwined with applications that handle personal data. By separating privacy policies from applications, privacy audits are simplified since they can focus on the privacy policies and usage logs that can be accessed via a privacy management system.

Note that this is only the technical core of privacy-enabled customer data management. Another important building block is to provide additional business processes that implement customer privacy services. Customers should be enabled to inspect and update the data and usage logs stored about them. In addition, an enterprise may offer the option to delete the personal data. Ideally, customers should retain maximum control over their data. Once privacy-management has been implemented, it needs to be audited by external parties that are trusted by the customers. Together with resulting privacy seals, this can increase the trust of the consumers.

For related work, we need to refer to our earlier work that survey different aspects of privacy management [1, 3, 5].

2 Application Model and Prerequisites

In our application model, an enterprise runs *legacy applications* that use collected data. Each application can perform certain *tasks*. For example, a "customer relationship management system (CRM)" application may

[2] Note that our scheme only protects against systematic privacy violations within the system. It cannot prevent misuse by an employee with legitimate access or misuse that falls outside the boundaries of the system. An example is that it cannot prevent an employee from copying information from the screen to a note pad.

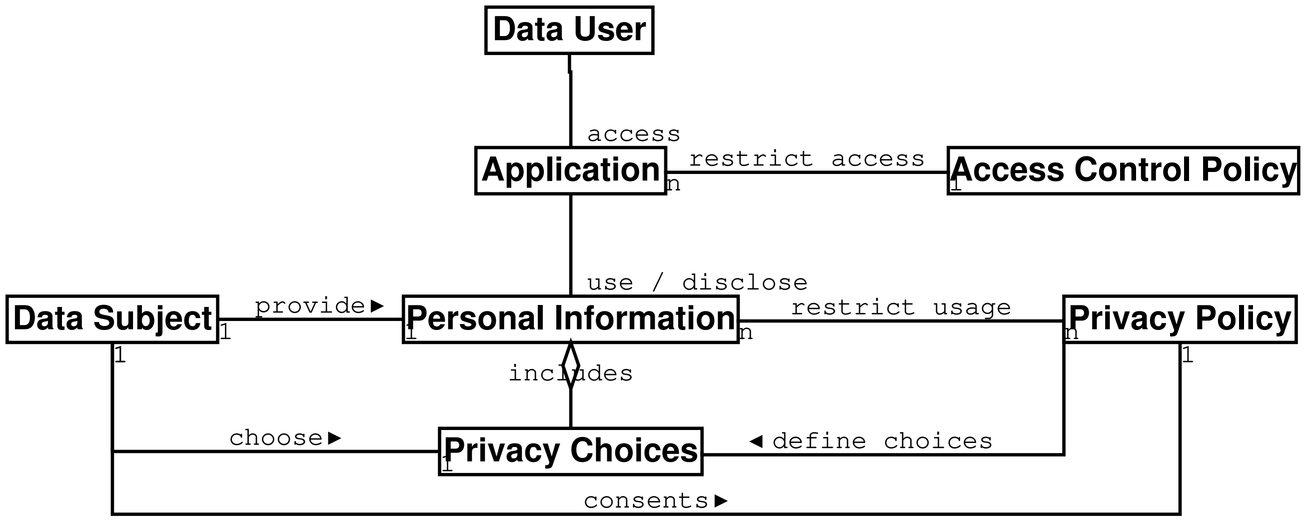

Figure 1: Separation of duties for privacy authorization.

perform the tasks "create new customer record" or "update existing customer record". Enterprise privacy policies reflect the authorized flow and usage of personal information within an enterprise. As a consequence, the flows and usages have to be identified in order to build a privacy enforcement system:

- A *business-process model* for the collection and use of customer data defines the scope of the data management system. The business-process model identifies the players that use collected data, the data they use, and how and for what purposes they use the data. The business model is formalized as the declaration of data, players and operations of an EPAL privacy policy.

- A collection of *informal privacy policies* that govern the use of personal data in the business processes. They can be structured as bilateral privacy agreements that describe how data that is sent from one player to another may be used. Informal privacy policies are formalized as EPAL privacy policy rules.

3 Defining Policies based on Separation of Duties

Privacy and security authorization in an enterprise involves at least four types of players. The *data subjects* are the players about whom personal data is collected. The most common data subjects are the customers of an enterprise. Other data subjects are employees or customers of cooperating enterprises. The next players are the *data users* within an enterprise who use collected data by executing *tasks* of *applications*. The other two players are the *privacy officer* (PO), who is responsible for privacy services, and the *security officer* (SO), who is responsible for security services.

Our privacy management architecture introduces the following intermediate abstractions and the corresponding policies (see Figure 1) in order to separate the duties of these players:

Privacy Policy The PO defines a *privacy policy*. A privacy policy describes what *operations* for which *purpose* by which *data user* can be performed on each *PII type*. For example, the "marketing department" may be allowed to "read" the PII type "contact data" for purpose "e-mail marketing". In addition, a privacy policy may define opt-in and opt-out choices for the data subjects as well as certain privacy obligations such as "delete my data after 30 days unless parental consent has been given".

The privacy policy should be enterprise- and application-independent. Enterprise-internals such as the role structure should not be used in order to enable exchange of policy-protected data between cooperating enterprises.

Deployment The PO and the SO define a *deployment policy* that maps legacy applications and their tasks onto the privacy-specific terminology used by the privacy policies. This mapping is specific to each enterprise. For example, whereas one enterprise maps a CRM system performing "product notification" as well as a printer for mass-mailings onto the action "read" for purpose "marketing", another enterprise, which uses a legacy application instead of the off-the-shelf CRM system, maps this legacy application onto "read" for "marketing".

The PO defines an *obligation mapping* that translates application-independent obligations of the privacy policy (such as "delete") into specific implementations. For example, a delete may be translated into an "unsubscribe" of the mailing list.

Access Control Policy The SO defines an *access control policy* that defines the roles and users within an enterprise. In addition, it defines which users or roles can execute which tasks of which applications.

Data Collection Personal data is collected at collection points. Each collection point has a set of fields, and fields have a type (e.g., string) as well as a PII type (e.g., "medical record", "address data", or "order data"). A form groups personal data and associates this data with its data subject. A "customer data" collection point, for example, may collect the fields "name", "street", and "town". Each collection point has a default policy that is assigned to data collected via this collection point. The system stores the data including opt-in or opt-out choices as well as a reference to the collection point that fixes the PII types and privacy policy.

4 Collecting Personal Data and Consent

When personal data is collected, it is core to add privacy management information that needs to be consented by the data subject. This main elements are the applicable privacy policy as well as the opt-in and opt-out choices of each individual data subject.

The collection catalog identifies the data and choices that are collected as well as the privacy policy and the PII types of the collected data. At a given collection point, the data subject enters its data in the fields of the given form. The collected data includes the fields and PII types of the entered data as well as a default policy. The data subject may then choose opt-in and opt-out choices defined by the policy. By submitting the form after reading the privacy statement, the data subject consents to the policy with respect to the selected choices. The choices are added to the form and the content of the form is stored.

Note that an enterprise privacy policy that can model access down to the employee level us usually too complex for end-users. As a consequence, it is advisable to present a coarser-grained privacy policy to the customer (either as text or P3P) and to define an EPAL policy for internal enforcement.

An important aspect is the management of the data subject's consent on a per-person and a per-record basis. This must be guided by the *sticky policy paradigm*: When submitting data to an enterprise, the user consents to the applicable policy and to the selected opt-in and opt-out choices. The form then associates the opt-in and opt-out choices as well as the consented policy with the collected data. This holds even if the data is disclosed to another enterprise. Note that policy management on a per-user basis is useful if consent and different sources are issues to be considered. Examples are managing data of different policy versions (e.g., due to different collection times), different user roles (e.g., paying users vs. users funded by advertising), or users from different jurisdictions (e.g., Europe and US).

5 Enforcing Privacy

At collection, the collected data has been associated with the privacy policy, and the selected privacy choices. This information is used to decide whether an access by an application shall be granted. Authorization is granted in two levels. Whereas access control focuses on restricting the access of employees to enterprise applications, privacy control restricts the access of applications to collected data.

Access Control: An employee acting as a data user with certain roles requests permission to perform a task of an application. The access control policy is used to verify that the data user with the given roles is in fact allowed to perform the requested task. If this is the case, the task is executed. This access control system is independent of the privacy authorization.

Privacy Control: Once a running task of a corresponding application has requested access to certain fields of collected data, the privacy enforcement system retrieves the form and uses it to allow or deny the given request as follows:

1. The request identifies the task of an application as well as the fields to be accessed.
2. The deployment maps the task onto a privacy-relevant operation and a purpose.
3. The collection point identifies the PII types of the requested fields.
4. The privacy policy and the data subject's choices are used to decide whether the operation for this purpose is allowed on the given PII types.
5. If the operation is denied, the access for the given task on the given fields is rejected.
6. If the operation is allowed and the privacy policy specifies a privacy obligation, the obligation mapping maps the obligation to a task of an application.[3]
7. If the operation is allowed, the task can be executed on the requested fields.

6 Components for Enterprise Privacy Enforcement

The authorization procedure described in Section 5 can be implemented by the privacy enforcement components depicted in Figure 2. The components interact as follows to decide whether a task executed by a legacy application is allowed to access a protected resource:

1. A *legacy application* tries to execute a task on a protected resource.
2. A resource-specific *resource monitor* shields the resource and captures the request of a certain task for certain fields. For each task, it asks the privacy management system for authorization.
3. The resource-independent *privacy management system* obtains an authorization query identifying the fields to be accessed by a certain task of a certain application. It performs steps 1 to 3 of the authorization procedure in Section 5 to deploy the authorization query.

[3] Applications are responsible for managing their data. As a consequence, they are required to implement tasks that correspond to obligations in the privacy policy.

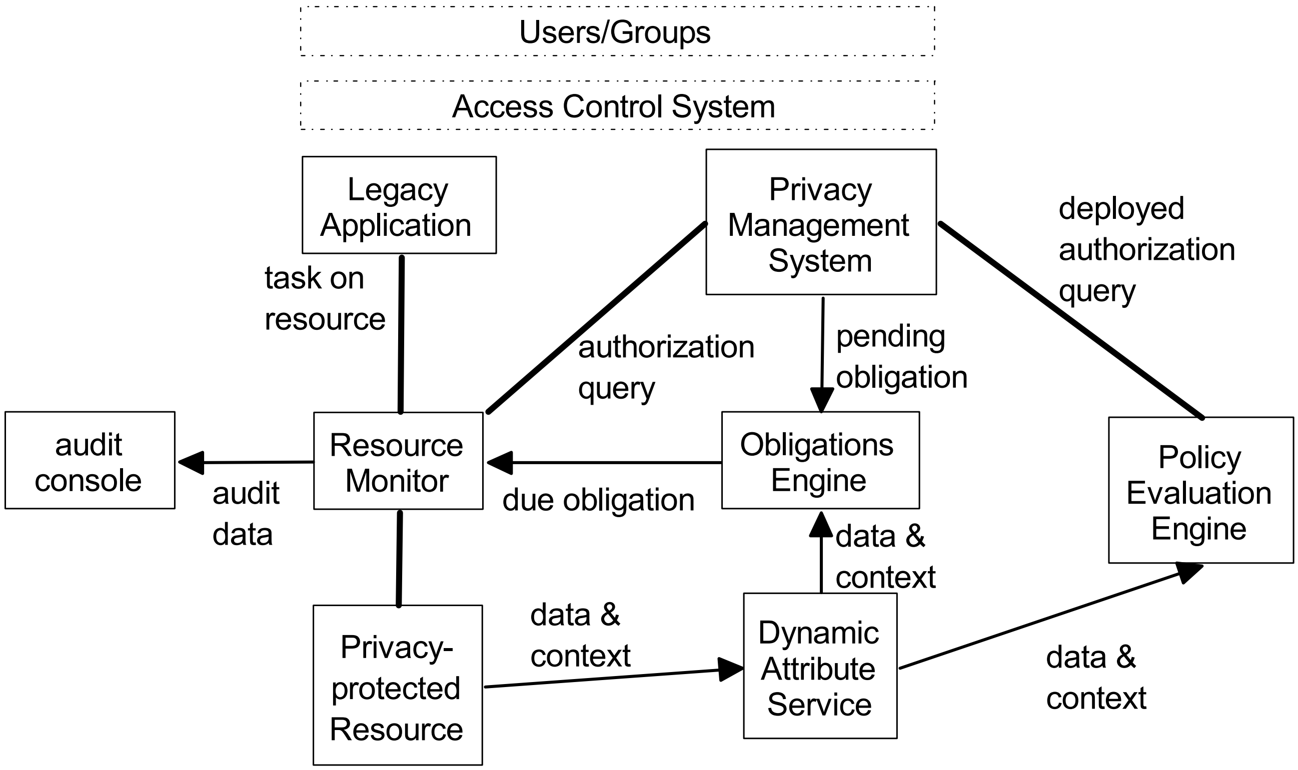

Figure 2: Architecture for privacy enforcement

4. The *policy evaluation engine* performs step 4 of the authorization procedure. The policy evaluation engine needs context and data to evaluate conditions. The resource monitor abstracts from resource and storage details by using a dynamic attribute service that provides values for data and context variables on request.

   The policy evaluation engine returns the decision as well as any resulting obligations to the privacy management system.

5. The *privacy management system* returns the decision of the policy evaluation engine to the resource monitor. If obligations were returned, the applications are mapped onto tasks (step 6) and sent to the obligations engine.

6. The *resource monitor* performs the tasks if it has been authorized. If not, the task is denied. In addition, the resource monitor sends log data to the audit monitor.

7. The resource-independent *obligations engine* stores all pending obligations. It evaluates the associated conditions based on values obtained from the dynamic attribute service. When a cancel-condition becomes valid, the obligation is removed. When a start-condition becomes valid, the obligation is sent to the resource monitor for execution.

7 Conclusion

We have described a comprehensive solution enterprise privacy management. EPAL enables an enterprise to formalize its privacy policy in an application-independent way. The deployment scheme enables enforcement of this common privacy policy for a variety of legacy systems. The viable separation of duties between the privacy officer and the security administrator enables secure and efficient management in practice. The intuitive consent-management paradigm enables customers to retain greater control over their personal data.

Our methodology enables an enterprise to protect personal data against misuse or unauthorized disclosure. It does not try to protect data if the enterprise systems or administrator are not trusted. Therefore, it merely augments a privacy-aware design of enterprise services that minimizes the data collected. In the desirable (but unlikely) scenario where an enterprise can offer its services without collecting personal data, our privacy management methodology would be rendered obsolete.

To correctly specify privacy rights and obligations that are being promised by privacy statements and mandated by a number of legislatures, the privacy officer must be able to reconcile easily what should be authorized with what is actually authorized. Therefore, we have developed a formal model for authorization management and access control in privacy protecting systems [4].

Acknowledgments

We thank Paul Ashley, Michael Backes, Kathy Bohrer, Nigel Brown, Jan Camenisch, Satoshi Hada, Calvin Powers and Els Van Herreweghen for valuable discussions and constructive comments.

References

[1] Michael Backes, Birgit Pfitzmann, Matthias Schunter: A Toolkit for Managing Enterprise Privacy Policies; 8th European Symposium on Research in Computer Security (ESORICS 2003), LNCS 2808, Springer-Verlag, Berlin 2003, 162-180.

[2] IBM: Enterprise Privacy Authorization Language (EPAL); Submission request to W3C, http://www.w3.org/Submission/EPAL/, November 2003.

[3] S. Fischer-Hübner: *IT-Security and Privacy*. Lecture Notes in Computer Science 1958, Springer-Verlag, 2001.

[4] G. Karjoth and M. Schunter. A Privacy Policy Model for Enterprises. In *15th IEEE Computer Security Foundations Workshop*. IEEE Computer Society Press, 2002.

[5] Günter Karjoth, Matthias Schunter, Michael Waidner: The Platform for Enterprise Privacy Practices - Privacy-enabled Management of Customer Data; 2nd Workshop on Privacy Enhancing Technologies (PET 2002), LNCS 2482, Springer-Verlag, Berlin 2003, 69-84.

[6] The Platform for Privacy Preferences 1.0 (P3P1.0) Specification, W3C Recommendation, 16 April 2002 (from www.w3.org/TR/2002/REC-P3P-20020416/).