Privacy and Identity Management in Europe for Life

First research report on research on next generation policies

Editors:	Pierangela Samarati (UNIMI)
Reviewers:	Ronald Leenes (TILT)
	Slim Trabelsi (SAP)

Identifier:	D5.2.1
Type:	Deliverable
Version:	1.0
Class:	Public
Date:	February 27, 2009

Abstract

This document describes the advancement status of the research work on policies in PrimeLife. It first highlights the overall objectives of Work Package 5.2 and then illustrates the main research results of the work package. The research results are related to the development of privacy-aware languages incorporating different cryptographic primitives and allowing the involved parties to define context-aware policies and privacy-aware constraints that regulate the data views accessible by the collaborating parties. Furthermore, an extensive analysis of the relationships between access control and data handling policies has been performed as well as an analysis of the main legal aspects related to the processing of personal data.

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 216483 for the project PrimeLife.

Members of the PrimeLife Consortium

1.	IBM Research GmbH	IBM	Switzerland
2.	Unabhängiges Landeszentrum für Datenschutz	ULD	Germany
3.	Technische Universität Dresden	TUD	Germany
4.	Karlstads Universitet	KAU	Sweden
5.	Università degli Studi di Milano	UNIMI	Italy
6.	Johann Wolfgang Goethe - Universität Frankfurt am Main	GUF	Germany
7.	Stichting Katholieke Universiteit Brabant	TILT	Netherlands
8.	GEIE ERCIM	W3C	France
9.	Katholieke Universiteit Leuven	K.U.Leuven	Belgium
10.	Università degli Studi di Bergamo	UNIBG	Italy
11.	Giesecke & Devrient GmbH	GD	Germany
12.	Center for Usability Research & Engineering	CURE	Austria
13.	Europäisches Microsoft Innovations Center GmbH	EMIC	Germany
14.	SAP AG	SAP	Germany
15.	Brown University	UBR	USA

List of Contributors

Contributions from several PrimeLife partners are contained in this document. The following list presents the contributors for the chapters of this deliverable.

Chapter	**Author(s)**
Chapter 1: Introduction	**UNIMI**
Chapter 2: Research results	**UNIMI**, EMIC, IBM, ULD, UNIBG
Chapter 3: Future Research	**UNIMI**, EMIC, IBM, ULD, UNIBG
Chapter 4: Conclusions	UNIMI
Chapter 5: Abstracts of research papers	**UNIMI**, UNIBG

Contents

List of Figures

Chapter 1

Introduction

Nowadays, a global information infrastructure connects remote parties worldwide through large scale networks, relying on application level protocols and services such as the World Wide Web. The vast amounts of personal information thus available has led to growing concerns about the privacy of their users. Users concerned about their private information are likely to refuse participation in such a global information infrastructure because they prefer not to be under the control of anyone at anytime. Furthermore, users also do not always realize that the information they disclose for one purpose (e.g., name, date of birth, and address within an on-line transaction) may also have secondary uses (e.g., access to existing data for grouping together users on the basis of common characteristics such as age or geographic location). Consequently, even if users agree to the initial collection of their personal information, they must also be provided with the possibility of specifying whether or not to consent to future uses of that information in secondary applications and under which conditions. This scenario poses an entirely new set of challenges to the design and implementation of privacy-aware models and languages, which is also complicated by the need to formally represent complex policies, where access decisions depend on the application (composition) of different rules (e.g., rules coming from laws practices, and organizational regulations).

Current policy languages allow the specification of regulations that establish who can, or cannot, execute which actions on which resources [SD01]. However, these traditional approach to access control policies result limiting in an open scenario aiming at privacy-awareness by the users. Although recent advancements allow the specifications of policies with reference to generic attributes/properties of the parties and the resources involved, they are not designed for enforcing privacy policies. The consideration of privacy issues introduces the need for rethinking authorization policies, models, and languages, and the development of new paradigms for access control and, in particular, authorization specification and enforcement.

A general goal of Work Package 5.2 is then the design of policy languages able to increase the level of user awareness, thus empowering the user with the ability to effectively control access and use by others over her data. The work in Work Package 5.2 has been organized into the following three tasks. *Task 5.2.1: Policy languages* focuses

on the definition of policy languages able to express sophisticated privacy needs (e.g., the specification and evaluation of context-based conditions and the enhanced treatments of certificates). *Task 5.2.2: Policies for web service composition* focuses on the security and privacy aspects related to the composition of Web services (e.g., the definition of a mechanism for enforcing a policy across multiple organizations). *Task 5.2.3: Legal policy mechanisms* focuses on the investigation of the protection models that are at the basis of current regulations.

In the first year of the project most attention has been devoted on Task 5.2.1 and Task 5.2.2; Task 5.2.3 has instead just started and has just began investigating the start of the art and existing approaches to define the work that will be performed in the next two years of the project. This consistently with the fact that very little resources have been employed for Task 5.2.3 in the first year of the project. The work performed by these tasks during the first year of the project can be summarized as follows.

- *Task 5.2.1* defined a privacy-aware language integrating cryptographic primitives, thus supporting anonymous credentials. Furthermore, this task has presented a first analysis of the privacy issues related to location-based services and has proposed a solution for regulating exceptions in health-care and for the specification and enforcement of privacy constraints in distributed environments.

- *Task 5.2.2* focused on the analysis of the relationships between access control and data handling policies. Research has been also done regarding the relationships between delegation of rights and secondary use and composition of policies.

- *Task 5.2.3* analyzed the principal legal aspects related to the management of personal data (e.g., transparency) and looked at the current status of this processing.

The remainder of this document is organized as follows. Chapter 2 presents the main research results obtained during the first year of PrimeLife. The chapter is organized into three sections, one for each task in the work package. Chapter 3 illustrates the issues that will be addressed in the remaining years of the project. Again, this chapter is organized into three sections, one for each task in the work package. Chapter 5 lists the abstracts of the research papers reporting the findings of the work package's partners. Finally, Appendix A is a glossary of the main terms used in this document.

Chapter 2

Research results

2.1 Policy languages (Task 5.2.1)

The work in this task has addressed the problem of analyzing existing policies languages and of developing new models, languages, and policies supporting complex privacy requirements emerging in different contexts (e.g., mobile, health-care, and database), where there is the need of integrating different sources of personal information. This problem has been studied from different perspectives, including the possibility of exploiting recent advances in cryptography that has permitted the use of anonymous credentials for developing privacy-aware languages. The novel contributions described in the following leverage on the research results of PRIME (www.prime-project.eu), especially on the PRIME privacy languages [ACDS08b], and have been developed with a common goal in mind: the need of having access control policy languages that, on one side, provide access control functionality and, on the other side, protect the privacy of the involved parties and of their personal information.

2.1.1 Exploiting cryptography for privacy-aware access control

Almost everyone uses electronic means for their daily interactions with businesses, governments, colleagues, friends, and family. In these interactions, users play different roles such as customer, citizen, patient, and family member and they disclose personal information ranging from attributes such as date of birth, age, and home address to credentials pertaining to skills and rights. Indeed, the number of transactions conducted electronically is ever growing and in fact not limited to those over the Internet as electronic authentication and authorization with some kind of token (e.g., electronic identity cards, driver's licenses, tickets and toll-tokens) become widespread.

Organizations endeavor to control access to the resources they provide based, among other things, on the attributes of the requestor. Access control thus is one of the key functions of identity management (IdM) from the perspective of enterprizes, although it is interweaved with other functions. Identity management systems in this context maintain digital identities, or accounts, containing attributes (e.g., a name) and properties (e.g.,

entitlements within the system's domain such as access rights) of the entities (usually individuals) within their domain. The accounts have an identifier (e.g., username) and one or more authenticators (e.g., a password). The individual's need for control over his information is often neglected in traditional IdM systems, even though the personal information they reveal is highly sensitive.

More recently a shift in focus of identity management can be witnessed from a strict perspective of enterprize-centric access control to resources towards a perspective that takes the interests of the individual into account. A number of identity management systems are available today, being standardized, or developed that are illustrative for this change in focus. These include the open source project Higgins, Microsoft's CardSpace, the web services standards, and the Liberty Alliance set of protocols.

Recent advances in cryptography have sparked a new generation of IdM systems based on anonymous credentials [Cha85, Bra99, CL01]. The basic concept has been known for quite some time: instead of revealing all attribute values encoded in the user's credentials, anonymous credentials allow the user to prove that she possesses valid credentials with attributes that satisfy some claim, without revealing any more information about the attributes than what is directly implied by the claim, however. More recently, more cryptographic tools have been developed that further extend the functionality of anonymous credentials, such as verifiable encryption [CS03] and limited spending of credentials [BCC04, CHK$^+$06]. These IdM systems need to be governed by specific privacy policies that must not only be stated, but also be enforceable by technical means. This holds for the access control policies (ACP), which state the conditions that a requestor must satisfy to gain access to a resource; for the data handling policies (DHP), which state how the requestor's information should be treated once it is revealed; as well as for the trust policies (TP), which state which authorities can be trusted to certify what type of information. On the other hand, the policy languages should be expressive enough to leverage the advanced features offered by the underlying technology. Given the fast pace at which new cryptographic tools are being developed and their technical complexity, designing such a language is a highly non-trivial task. A policy language that combines elements from access control, data handling, and trust policies, and that is the first to model anonymous credentials and a number of related cryptographic extensions has been designed and described in a publication that is currently under review. The proposed language has been integrated with the following cryptographic primitives.

- *Anonymous credentials.* Anonymous credentials can be thought of as digitally signed lists of attribute-value pairs that allow the owner of such a credential to prove statements about attribute values without revealing any more information about them than what is directly implied by the statement. By issuing a credential (i.e., by signing the list of attribute-value pairs), the authority certifies that the user satisfies the described attributes.

  Anonymous credentials allow the user and the verifier to engage in an interactive *selective-show* protocol during which the user proves that she owns a valid credential of which the attribute values satisfy some *claim*. The only information leaked about the attribute values however is that the claim holds true.

- *Pseudonymous identification.* When a user regularly accesses the same service, she

may not want to reprove at each visit that she qualifies for using the service, but may prefer to establish a permanent user account instead. To protect her privacy, she wants to do so under a pseudonym: it is bad enough that all her actions at this service now become linkable, she does not want them to become linkable to her actions across other services too. The server, on the other hand, may want to prevent users sharing their account information with others, thereby giving non-qualified users access to the service as well.

A pseudonymous identification scheme allows a user to derive from a single master secret *msk* multiple unlinkable cryptographic pseudonyms $nym_1, nym_2, \ldots$, and later authenticate herself by proving that she knows the master secret underlying one of the cryptographic pseudonyms. Of particular interest are those pseudonymous identification schemes that are compatible with an anonymous credential scheme, allowing the same master secret key *msk* to be encoded as an attribute in *all* the credentials belonging to one user, thereby preventing sharing of credentials.

- *Verifiable encryption.* A verifiable encryption scheme [CS03] is a public-key encryption scheme that is "compatible" with an anonymous credential scheme, in the sense that it allows to prove claims about how the encrypted content was derived from attributes in a credential—without revealing the content, however.

- *Limited spending.* Certain applications require that the number of times that a credential can be shown anonymously be limited [BCC04, CHK$^+$06]. For instance, a credential representing a wallet of n coins can be shown n times. A user can nevertheless attempt to use a credential more often. Cryptographic serial numbers[1] allow to detect overspending and, optionally, to obtain some information in escrow. The escrow is certified identifying information about the user that is hidden until an overspending occurs.

The policy language that integrates these cryptographic primitives is now presented by means of an example. Suppose that `Alice` considers to purchase a box of white wine at `CyberWinery.com`. Prior to the purchase, `Alice` creates an account at `CyberWinery`, thereby disclosing personal data. The account will store purchase data, personal preferences, and possibly even credit card data. `CyberWinery` has outsourced warehousing and delivery to 'LogisticsProvider', which requires data from `CyberWinery` (like a delivery address).

`Alice` proceeds with her purchase. Unlike many other web stores, `CyberWinery` lets `Alice` determine her own requirements for the data exchange. However, `CyberWinery` does require `Alice` to prove that she is creditworthy and over 18, but using *anonymous credentials* this is possible even within `Alice`'s choice to be *pseudonymous*.

After negotiating data handling policies, `Alice` consents to the disclosure of certain data compliant with her policies. When buying wine, `Alice` creates an encrypted token containing her address that can only be decrypted by LogisticsProcessor; there is no need for `CyberWinery` to know the delivery address.

[1] A cryptographic serial number looks like a random number, but is in fact deterministically derived from a unique *seed* embedded in a credential that can generate up to the *spending limit* different serial numbers.

	Access Control Policies	
	AC Rules	**Description**
ACP1	any WITH $\big($`identity_card`^{pk_1}[`age` $>$ 18, `nationality` $\in$ EU] $\vee$ `identity_card`^{pk_1}[`age` $>$ 21, `nationality` $\in$ non-EU]$\big)$ $\wedge$ *VerEnc*(*C*,pk_{s1}, `identity_card`^{pk_1}[`address`],"shipping") $\wedge$ `credit_card`^{pk_2}[`number`,`circuit`,`expiration`] $\wedge$ *VerEnc*(*C*,pk_{s2}, `credit_card`^{pk_2}[`name`],"failedpayment") $\wedge$ $\big($`identity_card`^{pk_1}[`name`] = `credit_card`^{pk_2}[`name`]$\big)\big)$ CAN `execute` ON `buy@CiberWinery` FOR `personal purchase`	A user is authorized to execute the `buy@CyberWinery` service for personal purchase purpose, if she owns a valid credit card, and she releases the identity card address, the credit card number, circuit, expiration, and name (name possibly encrypted), and she is European and older than 18 or she is non European and older than 21.
ACP2	any WITH `identity_card`^{pk_1}[`age` $>$ 16] CAN `browse` ON `CyberWinerySite` FOR `window shopping` IF `log_access`()	A user older than 16 can browse the `CyberWinery` Web site for window shopping purposes, if access is logged.
RP1	any WITH `business_card`^{pk_b}[`BBB_certified` = "yes"] CAN `access` ON `cc_info` WITH **object**.`expiration` $>$ *today* FOR `complete purchase`	A user is willing to give access to her valid credit card information only to BBB-certified entities for a complete purchase purpose.

Figure 1: An example of access control policies (ACP1 and ACP2) of `CyberWinery` and of an access control policy (RP1) of `Alice`.

Suppose now that `Alice` wants to buy a precious bottle of Italian red wine at `CyberWinery`'s website. To this aim, she submits a request of the form:

$$\langle Alice, execute, buy@CyberWinery, personal\ purchase \rangle.$$

The request is evaluated by the `CyberWinery` against the ACP in Figure 1. Based on *action*, *object*, and *purpose* in the request, ACP1 is the only applicable policy and the access request is evaluated against it. Let us suppose this is the first request by `Alice`, and then she is unknown to `CyberWinery` (i.e., her profile at `CyberWinery` is empty). The access evaluation result is "undefined" and `Alice` is prompted by `CyberWinery` with a request for additional information together with applicable DHP templates. `CyberWinery` asks `Alice` for the following set of information: *i)* a certification of the fact that she is European and older than 18 or non-European and older than 21; *ii)* a proof of possession of a credit card and the release of some attribute values in it, that is, *number*, *circuit*, *expiration*, *name* (attribute *name* can be released by means of a verifiable encryption); *iii)* a verifiable encryption containing the *address* to be used in the shipping process.

After receiving the request for information, `Alice` selects her applicable access control policies (RP1 in Figure1). Based on RP1, `Alice` is willing to release the data requested by `CyberWinery` if `CyberWinery` proves that it is a member of the Better Business Bureau (BBB). If this condition is verified, `Alice` releases data together with the DHP

	Data Handling Policies		
	PII	**DHP Rules**	**Description**
DHP1	`Alice.cc_info`	`business_card`^{pk_3}[`company`='CyberWinery'] CAN `read` FOR `complete purchase` PROVIDED `log_access`() FOLLOW `delete_after`(`purchase satisfied`)	An employee of `CyberWinery` can read the `cc_info` of `Alice` for complete purchase purposes provided that the access is logged. The data must be deleted after purchase is completed.
DHP2	`Alice.address`	`business_card`^{pk_3}[`company`='LogisticsProvider'] CAN `decrypt` FOR `shipping` FOLLOW `notify`(`Alice`)	An employee of `LogisticsProvider` can decrypt the `address` of `Alice` for shipping purposes. Data decryption must be notified to `Alice`.
DHP3	`Alice.name`	`business_card`^{pk_3}[`company`='CyberWinery'] CAN `decrypt` FOR `dispute resolution` PROVIDED `log_access`() FOLLOW `delete_after`(`6 months`)	An employee of `CyberWinery` can decrypt the `name` of `Alice` for dispute resolution purposes provided that the access is logged. Data must be deleted after six months.

Figure 2: An example of customized data handling policies that protect `Alice`'s data stored by `CyberWinery`

of Figure 2.

To complete the purchase process, `CyberWinery` needs to contact an external party, called `LogisticsProvider`, responsible for the shipping process. To send the wine to `Alice`, `LogisticsProvider` needs to decrypt the address information of `Alice`. Before any access is given to `Alice`'s data, the DHP in Figure 2 must be evaluated against `LogisticsProvider`'s request (i.e., $\langle$*LogisticsProvider,decrypt,Alice.address,shipping*$\rangle$).[2] The only applicable policy is DHP2, which evaluates to true. `LogisticsProvider` then decrypts the address information, sends the bottle of wine to `Alice`, and the relevant obligations are enforced. In our case, according to the obligations in DHP2, `Alice` must be notified about the access to her address data.

2.1.2 Privacy-aware languages empowering the users

The work has focused on the definition of novel solutions for guaranteeing awareness and empowerment of users in controlling access to their personal data. The work started with the study of the state of the art in privacy policy languages to see how (and whether) existing languages can be used to support the advanced features of privacy-aware systems [ACDS08b]. The conclusion seems to be that existing languages like XACML, P3P,

[2]In this case, also the ACPs of `CyberWinery` must be evaluated. For sake of conciseness, both in Figure 2 and in the discussion these additional ACPs are not described.

PRIME-AC (access control), and PRIME-DH (data handling) can certainly be useful as a basis, but they will need to be extended with a number of new concepts to fully appreciate the power of the technology. Three main lines of investigation have been then pursued: *i)* the definition of context (location)-aware privacy policies; *ii)* the definition of different policy spaces for supporting exceptions; *iii)* the specification and enforcement of privacy constraints in distributed scenarios.

Context (location)-aware privacy policies

Context information and, in particular, location-based information, should be used by the policy infrastructure to allow environment factors to influence how and when policy is enforced. The increasing availability of information about users's context makes it possible to develop context-sensitive services, where access to resources provided/managed by a server is limited depending on a user's context. For instance, a location-based service can require a user to be at a particular location to let the user use or access a resource or learn her friends' location. However, constraining access to a resource based on context information of users could result in privacy violations; if access is constrained based on the location of a user, granting or rejecting access will provide information about the location of the user and could therefore violate her privacy. Since the activities of a user are often related to the locations where such activities are performed, it is natural for users to demand privacy, that is, to require control over the access to their location information. Privacy issues in online and mobile services have been analyzed from different perspectives. Many sociological studies of the privacy problem [BD03, Col01] have brought to a better understanding of the concerns that users perceive when using a location-based service. From a technological point of view, instead, two different aspects of the current research on location privacy has been considered [BS04, BWJ05, GL08, HGXA07]: *1)* providing anonymity or support for partial identities to online and mobile services that are not based on the personal identification of a user for their provisioning; *2)* protecting the privacy of the location of the users, when user identity is needed for the successful service provisioning, by decreasing the accuracy of the location information itself. In general, in many real applications, location information can have sub-optimal accuracy levels and still offers an acceptable quality of service to end-users. As a consequence, the need for solutions that balance location accuracy and privacy protection arises.

The privacy aspects of using location information in *location-based services* (LBSs) [Fri08] have been analyzed and presented in [ACDS08a]. This analysis resulted in the identification of different categories of location privacy and in a classification of the main techniques that have been proposed to protect the location privacy. Furthermore, a privacy-aware LBAC architecture has been presented. Such an architecture is aimed at fully managing both LBSs and privacy requirements and includes four logical components:

- the *users*, whose location is captured through their mobile devices;
- a *location-based service* that requests location information of the users for a successful provisioning;
- a *location middleware* that interacts with different location providers and provides privacy-aware location services to the LBS (this is the component that explicitly

addresses the trade-off between location privacy and location accuracy by satisfying privacy preferences set by users and maximizing the quality of location information released to LBS);

- *location providers* that use location sensing technologies to provide location information.

The key component of this architecture is the location middleware that must effectively and securely manage the trade-off between accuracy and privacy.

Policy spaces for regulating exceptions

Another contribution has been the definition of an approach to regulate exceptions based on the concept of *policy spaces*. The proposal aimed at controlling and limiting the application of the *break the glass* approach used in practice to overrule the access control policy in emergency situations. This problem and a possible solution to it have been presented in a publication [ADG+08]. The paper has discussed the scenario of health-care systems that support interactions among patients, medical practitioners, insurance companies, and pharmacies. The very sensitive nature of the information managed by these systems requires the balance between two contrasting needs: the need for data, to guarantee a proper delivery of care, and the need for keeping data secure, to properly protect the privacy of patients. Access control is the base mechanism that health-care systems adopt for protecting medical data. Traditional access control models and policies are based on the assumption that the authorizations regulating access are known in advance. However, since in health-care systems an important requirement is that "nothing interferes with the delivery of care" [GD07], access control restrictions may need to be bypassed in case of emergencies and care delivery, especially when there is a risk for the patient's health. This *"break the glass"* phenomenon is an established pattern in health-care organizations and, though quite useful and mandatory in emergency situations, it represents a serious system weakness, for example, if breaking the glass becomes the norm [RE06]. The work in [ADG+08] has then presented an access control solution aimed at a better management of exceptions that occur in health-care. The proposed solution has been developed by having in mind the following requirements.

- The access control system should be designed to be flexible and extensible, and should not be limited to a particular model or language (depending on the context, different solutions might be utilized).
- The access control system should minimize the uncertainty by limiting those cases in which no regulation applies and the break the glass principle is used.
- The access control system should protect the privacy of the patients, and should not allow exchange of identity data that violates government legislations.

The proposed solution is based on the definition of different policy spaces regulating access to patient data and used to balance the rigorous nature of traditional access control systems with the prioritization of care delivery. Such a solution regulates the whole set of accesses, which would otherwise fall into a possible "break the glass" policy, and provides

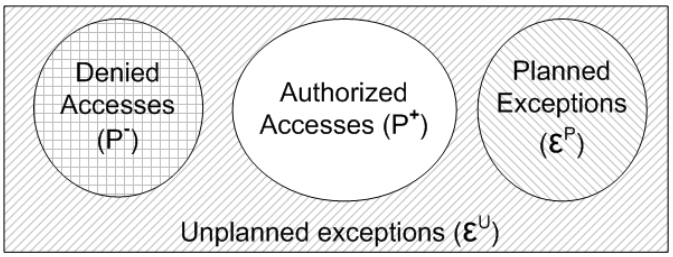


Figure 3: Policy spaces

a better treatment of "unusual" access requests. More precisely, the following set of *policy spaces* have been defined (see Figure 3).

- *Authorized Accesses.* It corresponds to traditional access control policies. Intuitively, this space includes the authorizations regulating *'common practice'*.
- *Denied Accesses.* It corresponds to access control policies that are used to prevent abuses. Denials are meant to be strictly enforced and do not allow any exception. They can be specified *a priori* to eliminate accesses that should never be authorized (i.e., accesses that cannot be bypassed by the break the glass) or inserted *a posteriori* because of observed abuses.
- *Planned Exceptions.* It corresponds to policies regulating access requests that do not fall into the normal routine as well as activities that should not be normally allowed. Policies in this space are evaluated if and only if there are no applicable policies in the previous spaces, or applicable policies have no effect. This space regulates access requests that are managed as exceptions and can be foreseen, for example, according to past observations.
- *Unplanned Exceptions*. It corresponds to policies regulating all access requests not covered by the previous policy spaces. Accesses regulated by this space are inserted into an auditing log for subsequent analysis and integration into the other spaces.

Even if no assumption has been made on the languages for specifying policies in the spaces, a language for planned exceptions has been proposed. The language supports the definition of two sets of rules: a set of *restrictions*, and a set of *authorizations*. Intuitively, *restriction* rules specify requirements that are *necessary* (but not sufficient) to have the request satisfied. By contrast, *authorization* rules specify permissions to be satisfied to have the access granted. Finally, the paper has illustrated how policies are specified and enforced within each space, and how these policy spaces are combined. To this aim, a policy evaluation and enforcement mechanism has been defined.

Specification and enforcement of privacy constraints in distributed scenarios

The last contribution has been the specification of privacy constraints that may arise in complex distributed and collaborative scenarios. This work is started on the observation

that often different sources storing the personal information of users need to collaborate to achieve a common goal. This collaboration, however, has to be carried out by taking into consideration the fact that different parties may be allowed to see different portions of the data [DFJ+08b]. The analysis of this problem has been performed by taking into consideration relational databases. This assumption is justified by the fact that relational databases have a central role in the management of large data collections today and promise to have an increasing role in future applications.

Current approaches for the specification and enforcement of authorizations in relational databases claim flexibility and expressiveness because of the possibilities of referring to views. Users can be given access to a specific portion of the data by the definition of the corresponding view (in the database schema) and the consequent granting of the authorization on the view to the user [Mot89, RMSR04, RS00, RS01]. It is then responsibility of the user to query the view itself. Queries on a table (base relation or view) are controlled with respect to authorizations specified on the tables themselves and are permitted only if the base tables are explicitly authorized. When the diversity of users and possible views is considerable and dynamic, such an approach clearly results limiting, because: *i)* it potentially requires to explicitly define a view for each possible access need and *ii)* it imposes on the user/application the burden of knowing and directly querying the view. In [DFJ+08a] we address these problems by proposing an expressive, flexible, and powerful, yet simple approach for the specification and enforcement of policies that can be applied in scenarios in which the diversity of users and possible views are considerable and dynamic. Such policies are based on the definition of authorizations that identify, in a declarative way, specific portion of the data whose access is being authorized. More precisely, authorizations express privileges not on specific existing views but on stable components of the database schema, exploiting both relations and joins between them; thus effectively identifying the specific portion of the data whose access is being authorized. A query on data distributed on different sources is then executed whenever the information carried by the query (either directly in the result or indirectly due to the dependence of the result with other data not explicitly released) is legitimate according to the specified authorizations. This is an important paradigm shift with respect to current solutions, departing from the need of specifying views to identify the portion of the data to be authorized but explicitly supporting such a specification in the authorizations themselves. From a modeling point of view, the proposed solution is based on the definition of query profiles capturing the information carried (directly or indirectly) by a query. The proposal builds then on a graph-based representation of the components of the authorization model (database schema, queries, and permissions) which is exploited for assessing if a query can be safely executed, that is, if the query is authorized by either an individual permission or a set of permissions that, taken in combination, provide the complete visibility of all information directly or indirectly carried by the query. Access control is then effectively modeled and efficiently executed in terms of graph coloring and composition and on traversal of graph paths. The composition of permissions is performed through a polynomial composition algorithm that avoids the computation of all possible permission compositions.

2.2 Policies for service composition (Task 5.2.2)

Task 5.2.2 investigates features required in privacy policy and preference languages to address service composition, e.g. workflows, mash-ups. The main focus is on composite services covering multiple trust domains. Data collected by such a composite service leads to new challenges in terms of data-flow control and data handling. Task 5.2.2 focuses on three main issues: *i)* the analysis of the relationships between access control policies (ACP) and data handling policies (DHP); *ii)* the analysis of the relationship between delegation of rights and data handling; *iii)* the investigation of mechanisms required to compose policies. The first and the second issue enable the definition and enforcement of policies within a composite service. The third issue aims at aggregating policies from different components of a composite service in order to define an overall policy. The remaining of this section describes first steps to address those issues.

2.2.1 Relationship between DH preferences and AC policies

Figure 4(a) shows a basic access control scenario, where a user gets data from a service. To compare AC with DH, we consider that the user acts as a data processor since he gets data and should handle it in an appropriate way. The service evaluates service's ACP and user's claims (e.g., Id, role, attributes) to decide whether access is granted or denied. Figure 4(b) shows a basic data handling scenario, where a data processor (service) needs personal data (PII) from data subject (user) to proceed. The user agent evaluates user's DH preferences to decide whether the personal data can be handed to the data processor. Service's claims (e.g., type of service, identity, and reputation) as well as service's DHP Template (e.g., purpose, commitment to fulfill obligations) is taken into account. Depending on the specific context, the DHP Template: *i)* cannot be customized (purely data processor driven such as with P3P) and thus the DHP does not have to be send back in step 3; *ii)* can be partially customized (e.g., PRIME DHP [ACDS08b]) because parts are optional or default values can be modified; *iii)* can be completely customized (i.e., data subject driven sticky policies where the data subject creates the DHP).

Consider now Figure 4(c) showing the same access control scenario as in Figure 4(a), where the data subject is a user and the data processor a service, and compare it with the basic DH scenario (Figure 4(b)). At a first glance, similarities between Figure 4(b) and Figure 4(c) are obvious: user agent and client's side AC engine have similar roles; Payload data data1 could be personal data; client side AC Policy and user's DH Preferences express similar constraints on the type of data processor that can access a specific personal data. However, there are the following important differences.

- In Figure 4(c), the AC engine only takes into account claims regarding the identity of the service but does not expect information on how data will be handled. As a result, there is no DHP attached to the personal data collected by the data processor, meaning that the AC engine returns a Boolean response (access granted or denied) while the user agent also processes the DHP template and returns a DHP.

- In Figure 4(b), the data subject (user) explicitly starts an interaction with the data

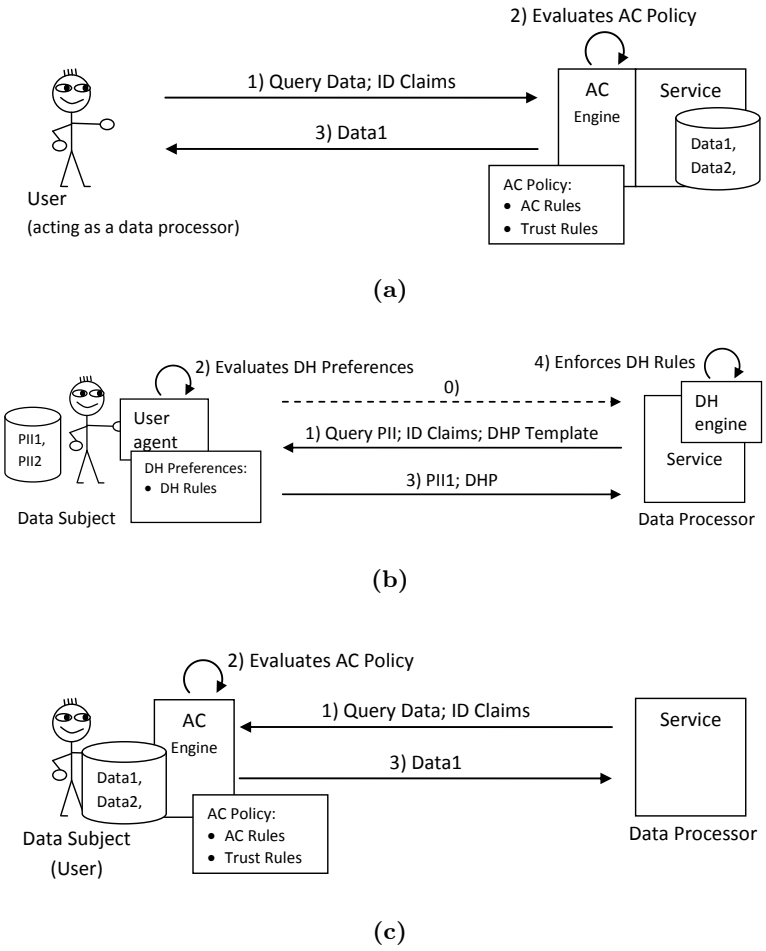


Figure 4: Basic access control (a), data handling (b), and backward access control (c) scenarios

processor (step 0). As a result, there is a better user control regarding who gets access to personal data.

- In Figure 4(c) the focus is on a specific aspect of ACP: read action on some data. ACP can generally be used to specify other types of actions (write, modify, delete, and so on).

- In Figures 4(a) and 4(c), the data subject keeps some control on his data. Indeed, assuming that data processors do not store the data, the data subject can forbid access at any time (e.g., by revoking credentials).

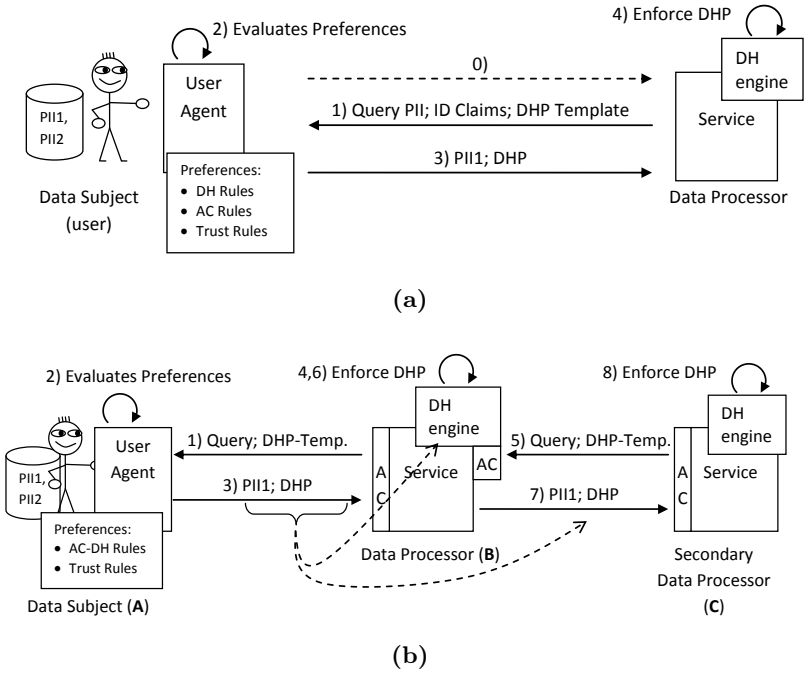


Figure 5: AC-DH combination with two (a) or three (b) parties involved

Combining ACP and DH Preferences

One of the main goal of Task 5.2.2 is to look at a superset of ACP and DHP enabling both types of scenario and adding features to ACP as well as to DHP. Features of some ACP languages (e.g., obligations in XACML) are already bridging part of the gap. Figure 5(a) presents an integration of both concepts. The user agent enforces rules related to trust (attributes of data processor and third parties, who can certifies attributes), access control (which personal information can be handed to a data processor with given attributes), and data handling (purpose of data collection, associated obligations). Similar rules may travel with the personal data (DHP in step 3) and are enforced at data processor side (step 4).

When data collected by a data processor is shared with a third party (secondary data processor), there are mainly the following two options.

- Access Control: the third party requests the personal data from the data processor and an access control decision is taken by this last one based on the DHP attached to the personal data. This option has been chosen in PRIME-DHP.
- Data Handling: the data processor hands the personal data to the third party with attached DHP. This requires dealing with the third party's policy, that is, verifying another DHP template provided by the third party to the data processor.

Covering both scenarios with a unique policy language would be valuable. Moreover,

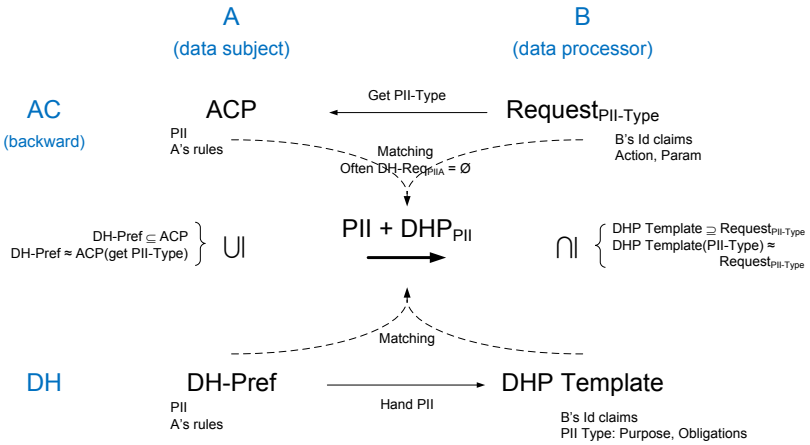


Figure 6: Relationship between access control and data handling

this would allow cases that are generally not supported: the data processor uses a service hosted by the data subject to get personal data (and associated DHP) and next share this data with third parties. Figure 5(b) shows how third parties can be tackled.

By making explicit the relationships between DH and AC (see Figure 6) and by analyzing DH preferences and AC policies, it is possible to conclude that there many similarities between DH and AC as well as many dissimilarities.

Similarities

- *Data subject side*: Access Control Policy (ACP) is similar to Data Handling Preferences ($DH\text{-}Pref$). Indeed both express under which condition A accepts to provide PII to B. ACP is however broader than $DH\text{-}Pref$. Indeed, A's ACP would define access rule for accessing A's PII but also other access rules for other services provided by A, e.g. who can call A, who can add entries to A's medical record.

- *Data processor side*: Data Handling Policy ($DH - Policy$) is similar to credentials and metadata attached to the query for PII ($Request_{PII-Type}$). Indeed both express which PII is required, trust information on B, and potentially why the PII is required and how it will be handled (including obligations). $DH - Policy$ is broader than $Request_{PII-Type}$ because it is not specific to the queried PII. Even in access control scenarios, it would make sense to have a data processor side (i.e. B-side) Policy to create $Request_{PII-Type}$.

- AC: Matching access request ($Request_{PII-Type}$) with AC rules (ACP) generally result in a Boolean authorization decision (granted or denied). In this case no data handling requirements (DHP_{PII}) are associated with data (i.e. PII, $DHP_{PII} = \emptyset$). However, in multiple cases, DHP_{PII} may contain additional data handling requirements such as obligations.

- *DH*: Matching service's data handling policy ($DHP - Template$) with user's data handling preferences (*DH-Pref*) generally result in a Boolean decision, i.e. whether A accepts to hand PII to B and a resulting data handling requirements "sticky data handling policy" (DHP_{PII}).

Dissimilarities

- Access Control is not a superset of Data Handling because AC does not address purpose and rarely addresses obligations. Moreover, AC generally does not require data owner "approval" when provided credentials match AC Policy. In DH scenarios, the user is generally aware that personal data is requested.

- Data Handling is not a superset of (backward) Access Control because DH only tackles scenarios where the data processor gets some data (PII_A) from the data subject. AC addresses broader scenarios where the action is not only "get" but also covers actions resulting in modifying data subject's data (PII) or not related to personal data at all.

- It would be valuable to have a language that unifies both AC and DH. We will investigate how existing AC languages (e.g., XACML, SecPAL) and/or DH languages (e.g., PRIME's DH-Policy) could be merged or extended to cover both AC and DH. Even if enforcement mechanisms are quite different, the policy could be expressed in a uniform way.

Hopefully, a common language offering as superset of usual AC and DH languages may cover both topics.

2.2.2 Relationships between Delegation of Rights and Data Handling

This section describes the links between delegation of rights in AC and data handling spanning multiple trust domains, i.e. when a data processor share collected data with a third party.

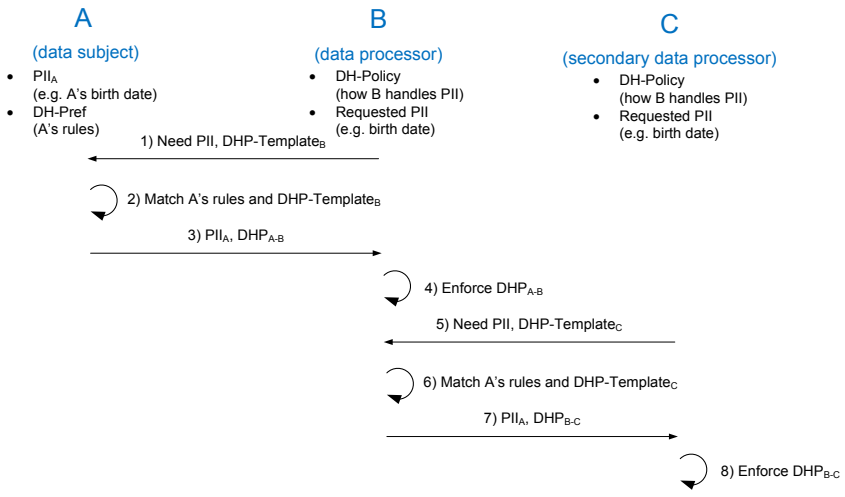
- *Delegation of rights in AC*: data subject (A, for short) authorizes data processor (B, for short) to access personal data (PII_A) and authorizes data processor (B) to delegate access rights to third party (secondary data processor C) so that C can access PII_A.

- *Sharing with third party in DH*: A provides to B some PII_A with attached constraints (DHP) on how to use PII_A and how to hand PII_A to a third party C.

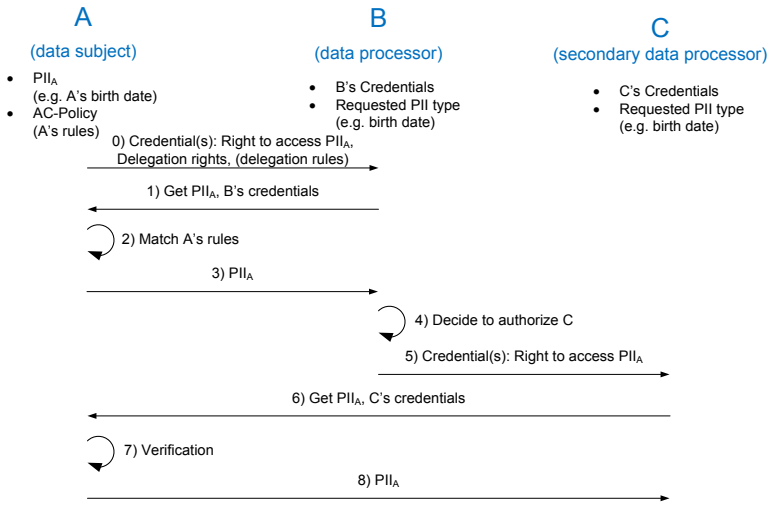
In both cases, 1) the data subject (A) may not know third party (C) in advance, 2) Data processor (B) decides whether third party (C) can get personal data (PII_A), 3) Data processor's (B) decision may be based on rules (DHP) provided by data subject (A).

Rules are provided by data subject and enforced by data processor:

- AC: B can delegate access to C for PII_A if some condition on C holds.



(a)



(b)

Figure 7: Conditional delegation in DH (a) and AC (b)

- *DH*: B can share PII_A with C if some condition on C holds.

In both cases, B is entitled to decide whether C can get PII_A based on conditions provided by A (but enforced by B). We call this "*conditional delegation*".

Figure 7(a) gives an overview of "conditional delegation" in DH setting. In this case, conditional delegation is used to let data processor share collected personal data with third parties.

1. B needs PII and describes in $DHP - Template_B$ how provided PII will be handled (including sharing with third parties)

2. A's preferences match B's $DHP - Template$. A decides to hand PII_A to B

3. A hands PII_A to B with attached policy DHP_{A-B}, i.e. constraints on PII usage and on third parties that can get PII.

4. B locally handles PII_A according to attached policy

5. B needs to provide PII_A to C. C describes in $DHP - Template_C$ how provided PII will be handled.

6. **Conditional delegation**: based on A's constraints (DHP_{A-B}) and C's attributes ($DHP - Template_C$), B decides to provide PII_A to C.

7. PII_A and DHP_{B-C} are sent to C.

8. C locally handles PII_A according to DHP_{B-C}

We do not describe here more static scenarios where data subject A knows third party C and can specify that data processor B can share personal data PII_A with third party C.

Figure 7(b) gives an overview of "conditional delegation" in AC setting. In this case conditional delegation is used to delegate access rights.

0. B is authorized to access PII_A and to delegate access to third party (e.g., C). Constraints on third parties could be added.

1. B accesses A's PII

2. A's AC rules are evaluated (similar to step 2 in Figure 7(a)).

3. A provides PII_A to B

4. **Conditional Delegation** (of rights): B decides to let C access PII_A. This could be based on constraints provided by A (similar to step 6 in Figure 7(a)).

5. B provides rights/credentials to C.

6. C access PII_A

7. The delegation (e.g., chain of credentials) is verified.

8. A provides PII_A to C

We do not describe here more static scenarios where A knows C and can grant access to C.

2.2.3 Composition of policies

Composition of policies is a key issue when aggregating personal data and when composing services. The remaining of this section describes requirements of policy composition and challenges to be addressed.

While mechanisms to evaluate whether a query fulfill a policy are well known, it is generally difficult to compare policies.

The following rule is applied: *manipulation of committed policies must ensure that, in the direction of the data flow, policies are equally or more restrictive.*

We say that a data processor B is committed to a policy DHP when B exposed DHP while collecting data (e.g., P3P) or when B committed to enforce provided policies (e.g., DHP is a sticky policy from data subject A). We say that a policy is more restrictive than another when there are fewer rights and more duties.

Example 1 (More restrictive internal) Data processor B commits to the following policy DHP_{PII_A} on personal data PII_A: "PII_A can be used for purposes {statistics, confirmation} and will be deleted within 6 months". Data processor B could internally enforce a more restrictive policy DPH'_{PII_A}: "PII_A can be used for purpose contact and will be deleted within 3 months".

We note $DPH'_{PII_A} \sqsubseteq DPH_{PII_A}$ (i.e., DPH'_{PII_A} is less permissive than DPH_{PII_A}).

Example 2 (More restrictive sticky policy) B aggregates different data with associated policies:

- A's e-mail address (PII_{A1}) and associated policy $DHP_{PII_{A1}}$: "PII_{A1} can be used for purpose statistics, confirmation and must be deleted within 6 months".
- A's phone number (PII_{A2}) and associated policy $DHP_{PII_{A2}}$: "PII_{A2} can be used for purpose confirmation, can be shared with C, and must be deleted within 12 months".

Aggregated data $PII_{A1,2}$ could have policy $DHP_{PII_{A1,2}}$: "$PII_{A1,2}$ can be used for purpose confirmation and must be deleted within 6 months".

We note $DHP_{PII_{A1,2}} = DHP_{PII_{A1}} \sqcap DHP_{PII_{A2}}$, where $\sqcap$ means "permissive intersection": the result is less permissive (i.e., more restrictive) than individual policies. $DHP_{PII_{A1,2}} \sqsubseteq DHP_{PII_{A1}}$ and $DHP_{PII_{A1,2}} \sqsubseteq DHP_{PII_{A2}}$.

Example 3 (Less restrictive frontend policy) Data processor B aggregates policies from third parties (secondary data processors) C and D:

- DHP_B: "PII can be used for purpose statistics and will be deleted within 3 months".
- DHP_C: "PII can be used for purposes contact and will be deleted within 3 months".
- DHP_D: "PII can be used for purpose statistics and will be deleted within 12 months".

The resulting policy of of B could be: DHP'_B: "*PII* can be used for purposes statistics, contact and will be deleted within 12 months". In other words, $DHP'_B = DHP_B \sqcup DHP_C \sqcup DHP_D$, where $\sqcup$ means "permissive union": the result is more permissive (i.e. less restrictive) than individual policies. $DHP'_B \sqsupseteq DHP_B$, $DHP'_B \sqsupseteq DHP_C$, and $DHP'_B \sqsupseteq DHP_D$.

Example 4 (No commitment leads to arbitrary policy) A aggregates different data with associated policies: "A's e-mail address can be used for purposes {statistics, confirmation} and must be deleted within 6 months" and "A's phone number can be used for purpose confirmation, can be shared with C, and must be deleted within 12 months". The policy of aggregated data does not have to be more restrictive than the policies of individual personal data because nobody did commit to related policies. In other words, the data subject can choose policies associated with each personal data (including aggregated data).

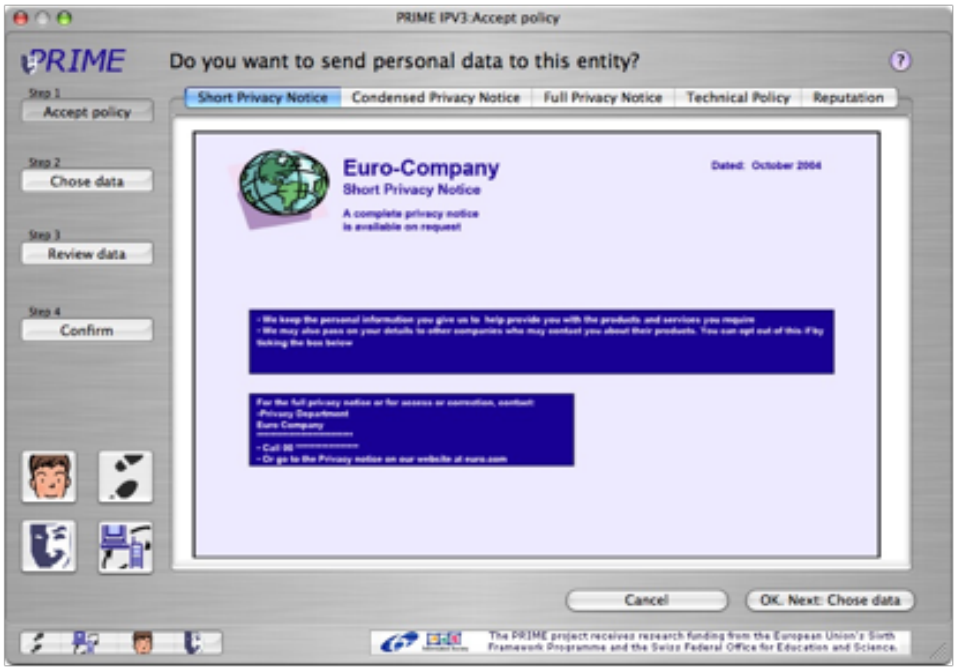
2.3 Legal policy mechanisms (Task 5.2.3)

Transparency is one of the core principles of data protection legislation in Europe [Eur95], beyond [APE05] and all around the world [OEC80]. The common understanding is that individuals should be aware of "who knows what about them" [BVe83]. This concept is supported by the principle of purpose or collection limitation[3], stating that any collected personal information must in principle only be processed for those purposes it was collected for. Roots of this principle lie in what Nissenbaum calls "contextual integrity" [Nis04], or the sociological concept of "functional differentiation" [Luh77]. Often enough these principles are hard to enact, enforce and above all understandable to the user. The user is confronted with plentitude of different purposes, often enough hidden in lengthy legal text of privacy notices (in the following: legal privacy policies), especially when surfing the web.

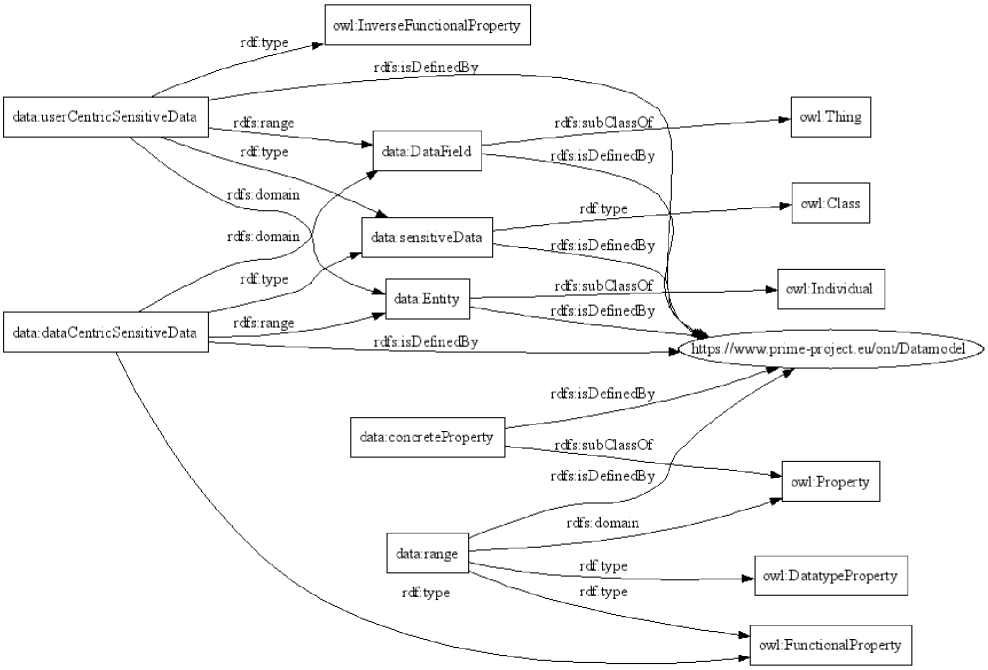
A number of approaches are currently trying to tackle this problem, by offering the user tools and mechanisms for a better understanding of what is happening with their data (see the discussion about prior work). However, most if not all of these approaches appear unclear what actually is necessary to communicate. The problem relates to the above. For a higher level of transparency, the user should be made aware about what actually happens to the data, who is processing it, and – if collected without active contribution of the user – what data is processed (e.g., Cookies, IP-Addresses, clickstreams, and so on). While the latter elements may be easy to communicate (but still might need some further thought), the question of how to express in a simple way, how the data are processed, for what purpose(s) they are collected poses difficulties. The multitude of applications and uses of personal data are highly unstructured, no comprehensive ontology exists, and no abstractions are apparent. For developing a typology of the processing of personal data an empirical approach, looking at the current practice seems appropriate.

The work in this task especially in the area of legal policy mechanisms aims at a better understanding of the legal aspects of the processing of personal data, by looking at the current status of this processing in different contexts and structuring these.

[3] [Eur95, APE05, OEC80], supra. See also [Sol06] for an outline of the relevance in the US-legal framework.



(a)



(b)

Figure 8: Prime HCI Mockup with multi-layered legal policies (a) and part of an ontology developed during the Prime project (b)

Recent and ongoing work in user transparency and legal privacy policies are made in the area of Human-Computer-Interfaces (HCI), as well as in technical representations and functional description of policies for privacy and data protection. The Article 29 Working Party has endorsed the use of multi-layered legal policies for web-sites [Par04], which has been implemented at several places on the web [Bro08]. Others have been developing tools and interfaces to support the user [Pet08]. Figure 8(a) illustrates an interface mockup developed during the PrimeProject. The interface allows seeing different levels of details of a privacy policy, following (and including) the example from the Annex to Art. 29 Working Party's Oppinion 100 (Art29WP04). Additionally a tab is included to access a technical expression of the privacy policy, which could be done in an enhanced version of P3P. Finally the tab on the far right offers information from third parties (labelled reputation). The concept also hints to the idea, that different uses of one set of data should be confirmed separately, in different steps (wizard-like-approach). Finally, there are proposals to use iconography to simplify recognition of a legal privacy policy for the user [Run06, Fis06] that have recently gained momentum [oP09].

For a formal description of privacy policies P3P is an available specification [W3C06], and offers some structural reference. XACML [OAS08], Liberty's Internet Governance Framework [Pro07], and WS-Policy[W3C07] are discussed for further expressing rules for the processing of personal data. However all these specifications lack or are very limited in offering conventions on describing the aspects most relevant to the user: What is actually going to happen to the data, what purposes are they used for, under what conditions and with what obligations?

Last but not least the Prime project [Kri08] did initiate some research for a more throughout ontology (see Figure 8(b) showing an approach to classify and structure elements of privacy policies) in this area and has reached to some first results, which further research should take into account, as well.

Chapter 3

Future research

The research results obtained in the first year of PrimeLife are important steps towards the main goal of Work Package 5.2. These research results are only the beginning and this work package will contribute with other research results. For each of the three tasks of the work package, we outline future work.

3.1 Policy languages (Task 5.2.1)

Considering the results presented in the previous chapter, the future work for this task will continue the investigation along the lines of work previously presented.

3.1.1 Exploiting cryptography for privacy-aware access control

The access control language proposed in [ACK+09] makes abstraction of the mathematical details of the cryptographic primitives, and focuses on their black-box behavior instead. Although this step is an important one in making the concepts understandable to non-cryptographers, it is not the final one. In particular, the predicates that were introduced in the policy language to model cryptographic pseudonyms, verifiable encryption, and cryptographic serial numbers, still concentrate more on the cryptography lying underneath, rather than the purpose that they serve. The legibility of the policy language could benefit greatly if it describes the goal that is envisioned, rather than the technology that is used to implement it. For instance, the predicate in the language that models verifiable encryption closely matches the input-output behavior of an actual verifiable encryption algorithm. It may be a better idea to introduce a separate predicate in the language for escrow, that is, giving an attribute (or combination of attributes) to a trusted third party that can be asked to reveal it in case of dispute, and a separate predicate that enables to specify *to whom* an attribute should be revealed. The latter would, for example, be used to specify that `Alice`'s address should be revealed to the `LogisticsProvider`, and not the `CyberWinery` itself.

The proposed language also takes a first step in the direction of seeing a credential as an atomic concept, a notion that is missing in popular languages like XACML. However,

if a user has multiple credentials of the same type by the same issuer, then the proposed language has no means to distinguish between these, or to specify that different attributes in the access control formula have to be encoded in the same predicate. For instance, if a concert ticket is a credential issued by TicketMaster, then you may have a first-row ticket to Jodlerklub Wiesenberg and a last-row ticket to U2. When the ACP insists that you show a credential for the valid concert and row, the current language cannot prevent you from obtaining a first-row spot to U2 by mixing the appropriate attributes from different credentials. In other situations, it may be necessary to insist that two credentials are different. New notation needs to be added to the language to support these features.

One can see that certain typical access control statements appear in the data handling clauses of the current language, and vice versa. For instance, rule ACP1 in Figure 1 of Chapter 2 mentions the purpose for which the data can be used, which is a typical data handling statement. Likewise, rule DHP1 in Figure 2 of Chapter 2 mentions that only requestors from company `CyberWinery` can obtain `Alice`'s credit card information, which sounds more like an access control restriction.

It is clear that ACPs, DHPs, and trust policies are tightly related concepts, but the relation among them is not very clear. More research is needed to distill the exact relation among these different types of policies, so that the architecture of a new language can be based on these new insights.

3.1.2 Privacy-aware languages empowering the users

Future work will address the following two main lines of research.

Privacy-aware techniques for location-based services

Although several proposals are aimed at supporting privacy in location-based services, they have some limitations. First, they do not provide a quantitative estimation of the provided privacy level, making them difficult to integrate into a full fledged location-based application scenario. Second, the location-based information is radically different from other context-related knowledge inasmuch it is both approximate (all location systems have a margin of error) and time-variant (location is subject to fast changes, especially when the user is in motion). Therefore, it is necessary to introduce a metric for measuring the relevance of location-based predicates evaluation. Third, even if there are many different types of techniques that addresses different privacy requirements, these techniques are always used in isolation. For instance, obfuscation techniques, which degrade the accuracy of the location information to provide privacy protection, are well-suited and can provide the required degree of protection. Unfortunately, the majority of the obfuscation-based proposals provide obfuscation by scaling (i.e., enlarging) the location area only.

We plan therefore to address these issues and to develop a novel solution aimed at preserving the location privacy of the users, by perturbing location information measured by sensing technologies. The solution proposed will be based on different obfuscation techniques used to protect a single sample of location information. A metric of both location information accuracy and privacy that abstracts from the physical attributes of the sensing technology as well as from the actual technique will be employed to obfuscate

a location. The obfuscation process will then allow, on the one hand, users to express their privacy preferences in a simple and intuitive way, and, on the other hand, to enforce the privacy preferences through a set of techniques (e.g., *enlarge,* to degrade the accuracy of a location measurement by enlarging its radius; *reduce,* to degrade the accuracy of a location measurement by reducing its radius; and so on). These techniques will be defined in terms of probability, and a solution to their composition will be provided. To validate the robustness of the solution, we plan also to define a threat model, where different adversaries will be analyzed based on the amount of their external knowledge.

Exception-based policies

We plan to extend our first proposal of a language for supporting exception-based policies by taking into consideration requirements and constraints coming from the applications (i.e., medical-like scenario with break the glass). The goal is to define a more general model and related language for regulating access in healthcare systems.

Security and privacy specifications

We plan to develop novel policy languages that will permit to identify under which conditions a party can trust others for their security and privacy. The exploitation of trust models represents a possibility that will be evaluated. Typically, such models are based on digital certificates (statements certified by given entities) can be used to establish properties of their holder (such as identity, accreditation, or authorizations). Although several approaches have been proposed for trust management and significant steps have been made in this direction, a major obstacle that still exists in the realization of the benefits of this paradigm is represented by the lack of adequate support in the DBMS. This is an important aspect to address since DBMSs are not only the backbone of old-style business applications, but are responsible for the management of most of the information that is accessed using a Web browser or a Web service invocation. We plan then to develop a trust model for DBMSs that will identify and adapt trust management concepts for their handling within relational databases.

3.2 Policies for web service composition (Task 5.2.2)

From the preliminary analysis conduced during the first year of the projects, it seems evident that a common language for AC and DH would be highly valuable to properly address data usage spanning multiple trust domains, i.e. when a data processor share collected data with a third party. This would be also valuable in *right management scenarios*. Rights Management Services (RMS) is a form of Digital Rights Management (DRM) used for protecting documents. With RMS, documents are encrypted and associated with a license. Let's assume that a data subject A sends an RMS protected document D_A to a data processor B. First, the data processor uses his credentials and the license provided by the data subject to get the decryption key. This is similar to AC where the data is the decryption key. Second, the enforcement of authorized actions on the document (print, listen ten times, and so on) are done at the data processor side. This is somehow comparable with purpose in DH policy. In other words, the document is

similar to personal data (PII_A) and the license is similar to a combination of AC policy (enforced by data subject) and DHP (enforced by data processor). We plan then to look at a superset of access control and data handling, focusing on matching preferences and policies. Also, another important aspect that will be addressed as future work is the specification of "conditional delegation" and its ability to let a data collector evaluate whether it is authorize to share collected data with a third party. We will then investigate how existing delegation of rights can be extended to cover data handling. Such an integration is required to cover cases where data handling and access control are mixed along the data flow.

Finally, with respect to policy composition, we will define more precisely policy operators such as "is more permissive than", "permissive union", and "permissive intersection". We will look at their impact on rights (use for purpose, share with third party, action, and so on) and duties (obligations, and so on). It is not clear whether such operators can be defined for existing languages such as XACML, PRIME-DHP, P3P, or SecPAL. Since those languages were not designed with this aspect in mind, we will certainly have to focus on less expressive subsets. This is well aligned with Work Package 5.2 overall goal to look at specific research aspects of data handling policy in parallel with the implementation of a more pragmatic approach in Work Package 5.3.

3.3 Legal policy mechanisms (Task 5.2.3)

For the empirical research, core parts of the ontology (Figure 8(b) in Chapter 2) were extracted, that are subject to further analysis. The goals are to find and to define, according to their relevance, the following aspects.

- Processes and services for personal data.
- Purposes of Processes and a partonomy/taxonomy thereof, to be sorted by relevance.
- Typical sets of data, expressed as a partonomy of data types and data sets.
- Data types and possibly qualifications/attributes (such as sensitive information as defined by 95/46/EC).
- Reference to the legal basis for the processing (norms, legal privacy policies).
- Text elements from legal privacy policies.
- Possible further elements, especially obligations, such as logging, deletion, blocking, information, retention periods (currently not included).
- Categories of data processors, and a partonomy/taxonomoy thereof (currently not included).

For this a database structuring the elements from the empirical research is to be filled as part of the work of the respective work package. A number of sources for this research are currently under evaluation, and will be approached step by step, while adopting structures and developing interfaces, allowing possibly further entities to contribute.

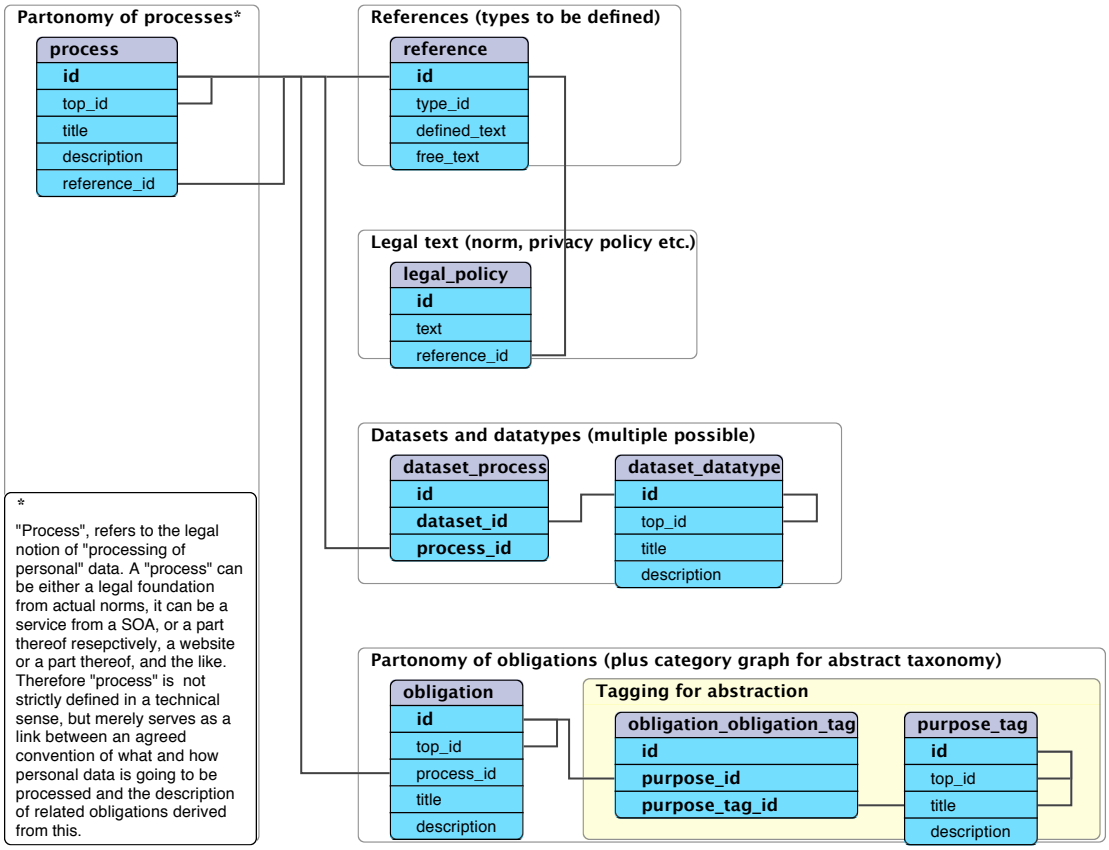


Figure 9: Preliminary annotated Entity-Relationship diagram of the underlying database for the empirical research

The empirical basis under evaluation comprises of:

- the German *corpus juris* especially those norms in federal law, allowing the processing of personal data by public bodies and agencies;
- "Verfahrensverzeichnisse", a speciality of German Data Protection Law, that might work as a good starting point;
- Privacy Statements from the Internet, possibly in cooperation with further entities;
- Service Oriented Architectures, possibly in cooperation with Activity 6;
- elements of the OECD Policy generator, and other similar prior work in the field;
- PrimeLife UseCases, to showcase the research.

Chapter 4

Conclusions

During the first year of PrimeLife, the work of Work Package 5.2 has been mainly focused on Task 5.2.1 and Task 5.2.2 while Task 5.2.3 has started the investigation on the state of the art to "trace the road" for the work that will be carried out in the next two years of the project. As testified by the publications produced, the work package has made significant contributions to the development of privacy-aware policy languages.

Task 5.2.1 has worked towards the definition of a policy language combining some features coming from access control, data handling, and trust policies, which is the first to model anonymous credentials and other related cryptographic extensions. Furthermore, Task 5.2.1 has developed novel models and languages for increasing awareness and empowerment of users in controlling access to their personal data. To this purpose, Task 5.2.1 has: 1) provided a first analysis of how to use location-based information to increase the expressive power of policy languages; 2) introduced the concept of policy spaces for supporting exception-based policies; 3) developed a novel model and efficient techniques for supporting and enforcing privacy constraints in distributed scenarios.

Task 5.2.2 has investigated the relationship between data handling policies and access control policies and has started the investigation on composition of policies. Although this task has no produced any publication, it has advanced the knowledge in the access control field and has provided interesting and novel ideas for the definition of a comprehensive framework supporting different privacy-aware policies.

Task 5.2.3 has provided a first analysis of the state of the art about the legal aspects of the processing of personal data.

The reported research results have been published in leading international journals and conferences (ACM CCS and Journal of Computer Security).
The research results presented in this report represent only a first step towards the realization of the overall goal of PrimeLife. The research will continue towards the development of novel and innovative models and languages for supporting users in protecting their privacy. Open issues that will be investigated include: the extension and enrichment of the proposed privacy-aware language by incorporating other novel cryptographic primitives; the development of novel models and languages for supporting the

"location privacy" of the users; the extension of the exception-based policies by taking into consideration requirements and constraints coming from the medical scenario; the definition of novel policy languages exploiting trust models for specifying under which conditions a party can trust another.

Chapter 5

Abstracts of research papers

5.1 Abstracts

1. C. A. Ardagna, M. Cremonini, S. De Capitani di Vimercati, P. Samarati, "Location privacy in pervasive computing," in *Security and Privacy in Mobile and Wireless Networking* [ACDS08a].
   Abstract. Recent technological advances have made it feasible to measure and track the location of users, vehicles, and practically any mobile object. Positioning and tracking systems are then collecting a huge amount of potentially sensitive *location information*, which is a set of data describing a user's location over a period of time. Since the activities of a user are often related to the locations where such activities are performed, it is natural for users to demand privacy, that is, to require control over the access to their location information.
   In this chapter, we focus on the privacy aspects of using location information in *location-based services* (LBSs). LBSs are services that take the current position of the user into consideration when performing their tasks. These services can be accessed from mobile phones, PDA, and any other mobile devices. We start the chapter by characterizing the location privacy protection problem and introducing a classification of the main techniques that have been proposed to protect the location privacy. We also survey and discuss recent proposals and ongoing work in the location-based systems area.

2. C.A. Ardagna, M. Cremonini, S. De Capitani di Vimercati, and P. Samarati, "A privacy-aware access control system," in *Journal of Computer Security (JCS)* [ACDS08b].
   Abstract. The protection of privacy is an increasing concern in our networked society because of the growing amount of personal information that is being collected by a number of commercial and public services. Emerging scenarios of user-service interactions in the digital world are then pushing toward the development of powerful and flexible privacy-aware models and languages. This paper aims at introducing concepts and features that should be investigated to fulfill this

demand. We identify different types of privacy-aware policies: access control , release, and data handling policies. The access control policies govern access/release of data/services managed by the party (as in traditional access control), and release policies govern release of personal identifiable information (PII) of the party and specify under which conditions it can be disclosed. The data handling policies allow users to specify and communicate to other parties the policy that should be enforced to deal with their data. We also discuss how data handling policies can be integrated with traditional access control systems and present a privacy control module in charge of managing, integrating, and evaluating access control, release, and data handling policies.

3. C.A. Ardagna, S. De Capitani di Vimercati, T. Grandison, S. Jajodia, P. Samarati, "Regulating exceptions in healthcare using policy spaces," in *Proc. of the 22nd Annual IFIP WG 11.3 Working Conference on Data and Applications Security* [ADG+08].
   Abstract. One truth holds for the healthcare industry - nothing should interfere with the delivery of care. Given this fact, the access control mechanisms used in healthcare to regulate and restrict the disclosure of data are often bypassed. This *"break the glass"* phenomenon is an established pattern in healthcare organizations and, though quite useful and mandatory in emergency situations, it represents a serious system weakness.
   In this paper, we propose an access control solution aimed at a better management of exceptions that occur in healthcare. Our solution is based on the definition of different policy spaces regulating access to patient data and used to balance the rigorous nature of traditional access control systems with the prioritization of care delivery.

4. S. De Capitani di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, P. Samarati, "Assessing query privileges via safe and efficient permission composition," in *Proc. of the 15th ACM Conference Conference on Computer and Communications Security (CCS 2008)* [DFJ+08a].
   Abstract. We propose an approach for the selective enforcement of access control restrictions in, possibly distributed, large data collections based on two basic concepts: *i)* flexible authorizations identify, in a declarative way, the data that can be released, and *ii)* queries are checked for execution not with respect to individual authorizations but rather evaluating whether the information release they (directly or indirectly) entail is allowed by the authorizations. Our solution is based on the definition of query profiles capturing the information content of a query and builds on a graph-based modeling of database schema, authorizations, and queries. Access control is then effectively modeled and efficiently executed in terms of graph coloring and composition and on traversal of graph paths. We then provide a polynomial composition algorithm for determining if a query is authorized.

Bibliography

[ACDS08a] C. A. Ardagna, M. Cremonini, S. De Capitani di Vimercati, and P. Samarati. Location privacy in pervasive computing. In S. Gritzalis, T. Karygiannis, and C. Skianis, editors, *Security and Privacy in Mobile and Wireless Networking*. Troubador Publishing, 2008.

[ACDS08b] C.A. Ardagna, M. Cremonini, S. De Capitani di Vimercati, and P. Samarati. A privacy-aware access control system. *Journal of Computer Security (JCS)*, 16(4):369–392, 2008.

[ACK+09] C.A. Ardagna, J. Camenisch, M. Kohlweiss, R. Leenes, G. Neven, B. Priem, P. Samarati, D. Sommer, and M. Verdicchio. Exploiting cryptography for privacy-enhanced access control. Manuscript in submission, 2009.

[ADG+08] C.A. Ardagna, S. De Capitani di Vimercati, T. Grandison, S. Jajodia, and P. Samarati. Regulating exceptions in healthcare using policy spaces. In *Proc. of the 22nd Annual IFIP WG 11.3 Working Conference on Data and Applications Security*, London, U.K., July 2008.

[APE05] APEC. APEC Privacy Framework, 2005.

[BCC04] E.F. Brickell, J. Camenisch, and L. Chen. Direct anonymous attestation. In *Proc. of the 11th ACM Conference on Computer and Communications Security (CCS 2004)*, Washington, DC, USA, October 2004.

[BD03] L. Barkhuus and A. Dey. Location-based services for mobile telephony: a study of user's privacy concerns. In *Proc. of the 9th IFIP TC13 International Conference on Human-Computer Interaction (INTERACT 2003)*, Zurich, Switzerland, September 2003.

[Bra99] S. Brands. *Rethinking Public Key Infrastructure and Digital Certificates — Building in Privacy*. PhD thesis, Technical University Eindhoven, 1999.

[Bro08] K. Brown. The infocard identity revolution, 2008. http://technet.microsoft.com/en-us/magazine/cc160966(printer).aspx.

[BS04] A.R. Beresford and F. Stajano. Mix zones: User privacy in location-aware services. In *Proc. of the 2nd IEEE Annual Conference on Pervasive Computing and Communications Workshops (PERCOMW04)*, Orlando, FL, USA, March 2004.

[BVe83] BVerfG. Volkszählungsurteil. *Bundesverfassungsgericht der Bundesrepublik Deutschland*, 65:1, 1983.

[BWJ05] C. Bettini, X.S. Wang, and S. Jajodia. Protecting privacy against location-based personal identification. In *Proc. of the 2nd VLDB Workshop on Secure Data Management*, Trondheim, Norway, 2005.

[Cha85] D. Chaum. Security without identification: Transaction systems to make big brother obsolete. *Communications of the ACM*, 28(10):1030–1044, 1985.

[CHK+06] J. Camenisch, S. Hohenberger, M. Kohlweiss, A. Lysyanskaya, and M. Meyerovich. How to win the clonewars: efficient periodic n-times anonymous authentication. In *Proc. of the 13th ACM Conference on Computer and Communications Security (CCS 2006)*, Alexandria, VA, USA, October-November 2006.

[CL01] J. Camenisch and A. Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In Birgit Pfitzmann, editor, *Advances in Cryptology – EUROCRYPT 2001*, volume 2045 of *Lecture Notes in Computer Science*, pages 93–118. Springer, 2001.

[Col01] M. Colbert. A diary study of rendezvousing: implications for position-aware computing and communications for the general public. In *Proc. of the International 2001 ACM SIGGROUP Conference on Supporting Group Work (GROUP 2001)*, Boulder, Colorado, USA, September-October 2001.

[CS03] J. Camenisch and V. Shoup. Practical verifiable encryption and decryption of discrete logarithms. In Dan Boneh, editor, *Advances in Cryptology – CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 126–144. Springer, 2003.

[DFJ+08a] S. De Capitani di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Assessing query privileges via safe and efficient permission composition. In *Proc. of the 15th ACM Conference Conference on Computer and Communications Security (CCS 2008)*, Alexandria, Virginia, USA, October 2008.

[DFJ+08b] S. De Capitani di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Controlled information sharing in collaborative distributed query processing. In *Proc. of the 28th International Conference on Distributed Computing Systems (ICDCS 2008)*, Beijing, China, June 2008.

[Eur95] 95/46/EC (Data Protection Directive), 1995. European Commission.

[Fis06] J. Fishenden. Creative commons – and its wider potential, 2006. http://ntouk.com/?view=plink&id=135.

[Fri08] L. Fritsch. Profiling and location-based services (lbs). *Profiling the European Citizen*, pages 147–168, 2008.

[GD07] T. Grandison and J. Davis. The impact of industry constraints on model-driven data disclosure controls. In *Proc. of the 1st International Workshop on Model-Based Trustworthy Health Information Systems (MOTHIS) 2007*, Nashville, Tennessee, USA, September 2007.

[GL08] B. Gedik and L. Liu. Protecting location privacy with personalized k-anonymity: Architecture and algorithms. *IEEE Transactions on Mobile Computing*, 7(1):1–18, January 2008.

[Hea] *Health Insurance Portability and Accountability Act*. `http://www.dhhs.gov/ocr/hipaa/`.

[HGXA07] B. Hoh, M. Gruteser, H. Xiong, and A. Alrabady. Preserving privacy in GPS traces via density-aware path cloaking. In *Proc. of the ACM Conference on Computer and Communications Security (CCS)*, Alexandria, VA, USA, October 2007.

[Kri08] T. Kriegelstein. *PRIME Data Model*. PRIME Consortium, 2008. https://www.prime-project.eu/ont/Datamodel.html.

[Luh77] N. Luhmann. Differentiation of society. *Canadian Sociological Review*, 2:, S. 29–53, 1977.

[Mot89] A. Motro. An access authorization model for relational databases based on algebraic manipulation of view definitions. In *Proc. of the ICDE89*, Los Angeles, CA, February 1989.

[Nis04] H.F. Nissenbaum. Privacy as Contextual Integrity. *SSRN eLibrary*, 2004.

[OAS08] OASIS. OASIS eXtensible Access Control Markup Language (XACML) TC, 2008.

[OEC80] OECD. *OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data*. Organisation for economic Co-operation and Development, 1980.

[oP09] Dynamic Coalition on Privacy. Privacy-rights-agreements, 2009.

[Par04] Article 29 Working Party. Opinion on more harmonised information provisions, 2004.

[Pet08] J. Sören Pettersson. *HCI-Guidelines*. PRIME Consortium, 2008.

[Pro07] Liberty Alliance Project. Identity governance, 2007.

[RE06] L. Rostad and O. Edsberg. A study of access control requirements for healthcare systems based on audit trails from access logs. In *Proc. of the 22nd Annual Computer Security Applications Conference on Annual Computer Security Applications Conference (ACSAC 2006)*, Miami Beach, Florida, USA, December 2006.

[RMSR04] S. Rizvi, A. Mendelzon, S. Sudarshan, and P. Roy. Extending query rewriting techniques for fine-grained access control. In *Proc. of the SIGMOD 2004*, Paris, France, 2004.

[RS00] A. Rosenthal and E. Sciore. View security as the basis for data warehouse security. In *Proc. of DMDW'2000*, Stockholm, Sweden, June 2000.

[RS01] A. Rosenthal and E. Sciore. Administering permissions for distributed data: factoring and automated inference. In *Proc. of the IFIP 11.3 Working Conference in Database Security*, Niagara, Ontario, Canada, July 2001.

[Run06] M. Rundle. International data protection and digital identity management tools (using icons to express user preferences), 2006. http://identityproject.lse.ac.uk/mary.pdf.

[SD01] P. Samarati and S. De Capitani di Vimercati. Access control: Policies, models, and mechanisms. In R. Focardi and R. Gorrieri, editors, *Foundations of Security Analysis and Design*, volume 2171 of *LNCS*. Springer-Verlag, 2001.

[Sol06] D.J. Solove. A Taxonomy of Privacy. *GWU Law School Public Law Research Paper No. 129*, 2006. http://ssrn.com/paper=667622.

[W3C06] The Platform for Privacy Preferences 1.1 (P3P1.1) Specification, 2006. http://www.w3.org/TR/P3P11/.

[W3C07] Web Services Policy 1.5 - Framework, 2007. http://www.w3.org/TR/ws-policy/.

A. Glossary

This glossary has been compiled from the Wiki page: https://trac.ercim.org/primelife/wiki/WP5.2-AC-DH.

Access Control Policy (ACP). Description of who can do a given action on a given object.

Access Request (and Claims). Description of a required action on a given object and associated claims on the requester.

Data Handling Policy (DHP). Description of how data can be used and shared. DHP-Template: Partially defined DHP. DHP is created from DHP-Template by filling missing parts.

Data Handling Preferences. Description of user expectation regarding how personal data can be used and shared.

Data Processor. Party consuming personal data (primary use). Secondary Data Processor (or Third Party): every party not involved in the initial agreement between two or more parties.

Data Subject. The individual or thing that has a governance relationship to the personal data and is the ultimate decision holder. Decisions may be delegated, but this is the instance one would have to return back for decision in case of doubt. For the sake of simplicity, in this document, we assume that the data subject is the user of the system.

Personal Data. Personal Data is the object that data handling directives act upon. For instance, IP address, first name, age, browsing history, X-ray, or current location can be considered as personal data.

Policy Engine. Piece of code running enforcing policies (e.g., DHP or ACP). User Agent: Piece of code running under the control of data subject. Its main task is to enforce data subject's preferences when PII is collected.