Partage et Contrôle d'Usage de Données Personnelles

Athanasia Katsouraki

NNT : **2016SACLV089**

THESE DE DOCTORAT
DE
L'UNIVERSITE PARIS-SACLAY
PREPAREE A
L'UNIVERSITE DE VERSAILLES SAINT-QUENTIN EN YVELINES

ÉCOLE DOCTORALE N°580 STIC
Sciences et technologies de l'information et de la communication

Spécialité de doctorat : Informatique

Par

Mme Katsouraki Athanasia

Sharing and Usage Control of Personal Information

Thèse présentée et soutenue à Versailles, le 28 Septembre 2016 :

Composition du Jury :

M. Philippe BONNET, Professeur, ITU Copenhague, Président
M. Philippe BONNET, Professeur, ITU Copenhague, Rapporteur
M. Sébastien GAMBS Professeur, Université du Québec à Montréal, Rapporteur
M. Matthieu MANANT , Maître de Conférences, Université de Paris Sud, Examinateur
M. Jean Marc PETIT, Professeur, INSA Lyon, Examinateur
M. Luc BOUGANIM, Directeur de Recherches, INRIA Saclay - Île-de-France, Directeur de thèse
M. Benjamin NGUYEN, Professeur, INSA Centre-Val de Loire, Co-directeur de thèse

All of science is nothing more that the refinement

of everyday thinking.

- Albert Einstein

UNIVERSITÉ DE VERSAILLES ST-QUENTIN-EN-YVELINES

université PARIS-SACLAY

Titre : Partage et Contrôle d'Usage de Données Personnelles (en français)
Mots clés: Privacy, Security, Access Control, use control, workflows, Operators.

Résumé: Nous vivons une véritable explosion du volume de données personnelles numériques qui sont générées dans le monde chaque jour (ex. capteurs, web, réseaux sociaux, etc.). En conséquence, les particuliers se sentent exposés lorsqu'ils partagent et publient leurs données. Ainsi, il est clair que des outils et des méthodes sont nécessaires pour contrôler la façon dont leurs données sont collectées, gérées et partagées. Les défis sont principalement axés sur le manque d'applications ou de solutions techniques qui assurent la gestion et le partage sécurisés de données personnelles. Le défi principal est de fournir un outil sécurisé et adaptable qui peut être utilisé par tout utilisateur, sans formation technique. Cette thèse fait trois contributions dans le domaine de la protection de la vie privée : (i) Une implémentation du model $UCON_{ABC}$, un modèle de contrôle d'usage, appliqué à un scénario de réseau social, (ii) une extension algébrique de UCON pour contrôler des partages complexes de données (en transformant des données personnelles en données partageable et/ou publiables), et (iii) la conception, l'implémentation et le déploiement sur le terrain de deux plateformes. La première, utilise dans le cadre d'un projet pédagogique afin d'aider des étudiants à développer des applications ''Privacy-by-Design''; la seconde pour la gestion de données sensibles collectées au travers de formulaires d'enquêtes.

Title : Sharing and Usage Control of Personal Information
Keywords : Privacy, Security, Access Control, use control, workflows, Operators.

Abstract: We are recently experiencing an unprecedented explosion of available personal data from sensors, web, social networks, etc. and so people feel exposed while they share and publish their data. There is a clear need for tools and methods to control how their data is collected managed and shared. The challenges are mainly focused on the lack of either applications or technical solutions that provide security on how to collect, manage and share personal data. The main challenge is to provide a secure and adaptable tool that can be used by any user, without technical background. This thesis makes three contributions to the field of privacy: (i) a prototype implementation of the $UCON_{ABC}$ model, a usage control model, applied to an online social networks scenario, (ii) an algebraic extension to UCON to control the complex sharing of data (by transforming personal data into sharable and publishable data) and (iii) the design, implementation and field testing of two secure platforms the first one being used in teaching class laboratories to help students designing Privacy-by-design applications, the second one to manage sensitive data collected through online forms.

Résumé étendu de la thèse

Nous vivons une véritable explosion du volume de données personnelles numériques qui sont générées dans le monde chaque jour (ex. capteurs, web, réseaux sociaux, etc.). Par exemple, dans le cadre des réseaux sociaux, les utilisateurs souhaitent partager leurs informations qui incluent photos, vidéos, messages texte, sondages, etc. Le processus de partage de l'information et la diffusion dépendent du réseau social, mais en général, un utilisateur publie des informations, visibles par ses amis qui peuvent alors décider de les republier.

Ces informations sont souvent peu structurées: certaines informations, que nous appellerons des données, sont d'un type spécifique, tel qu'une image ou une photographie. Ces données sont enrichies par des métadonnées, telles que l'emplacement de la photo, date/heure, ou la liste des personnes qui étaient sur l'image. Les métadonnées peuvent être générées automatiquement (ex. par l'appareil photo lorsque la photo est prise), ou rajoutées manuellement par les utilisateurs eux-mêmes, un exemple typique étant le marquage d'amis sur les photos sur Facebook.

En conséquence, les particuliers se sentent exposés lorsqu'ils partagent et publient leurs données. Ainsi, il est clair que des outils et des méthodes sont nécessaires pour contrôler la façon dont leurs données sont collectées, gérées et partagées. Les défis sont principalement axés sur le manque d'applications ou de solutions techniques qui assurent la gestion et le partage sécurisé de données personnelles. Le défi principal est de fournir un outil sécurisé et adaptable qui peut être utilisé par tout utilisateur, sans formation technique.

Cette thèse a été réalisée dans un contexte multidisciplinaire dans le cadre de l'Institut de la Société Numérique (ISN) de Paris-Saclay et de son groupe de travail sur la vie privée, qui inclut des juristes, des économistes et des informaticiens. Ainsi, cette thèse porte sur la question du modèle de contrôle d'usage UCON à travers deux axes: (a) l'informatique et (b) la socio-économie. L'axe informatique couvre les algorithmes, les modèles et la conception du système, et l'axe socio-économique couvre une expérience terrain avec de vrais utilisateurs, pour mesurer la facilité d'utilisation et la capacité d'adaptation des utilisateurs à ces nouveaux systèmes.

Précisément, cette thèse fait trois contributions dans le domaine de la protection de la vie privée :
(i) Une implémentation du model UCON_{ABC}, un modèle de contrôle d'usage, appliqué à un scénario de réseau social. Brièvement, le contrôle d'usage (UCON) est un moyen de spécifier comment un utilisateur ou un producteur de données peuvent contrôler les usages de celles-ci. UCON gère l'accès aux données, mais aussi ce qui peut être fait avec ces données. Le modèle UCON est primordial dans le contexte des

réseaux sociaux, où les utilisateurs souhaitent partager leurs données, mais aussi conserver un certain contrôle, notamment lorsque ces dernières sont disséminées.

(ii) une extension algébrique de UCON pour contrôler les partages complexes de données (en transformant des données personnelles en données partageable et/ou publiables). Elle permet à un utilisateur de spécifier les workflows de confidentialité préservant leurs données. Une originalité de cette approche est que ces workflows peuvent être combinés avec d'autres workflows, ou exécutés par d'autres utilisateurs sur leurs propres données.

(iii) la conception, l'implémentation et le déploiement sur le terrain en deux plateformes. La première plateforme est simple (blocs de base minimum), utilisée dans le cadre d'un projet pédagogique afin d'aider des étudiants à développer leurs propres applications ''Privacy-by-Design''. Cette plateforme a été testée par environ cent cinquante élèves, à l'ENSIEE en 2015 et à l'ENSIEE, l'UVSQ, et l'INSA en 2016. La seconde plateforme est utilisée pour la gestion de données sensibles collectées au travers de formulaires d'enquêtes. Cette plateforme a été testée dans le cadre des projets PAIP et Valdo par cent quarante utilisateurs, sans formation technique et elle sera réutilisée par les économistes du laboratoire RITM à travers un "laboratoire expérimental mobile", dans le but de recueillir des statistiques sur des données personnelles à des fins de recherche, tout en protégeant la vie privée des utilisateurs.

Acknowledgements

First and foremost I would like to express my sincere gratitude to my supervisor Prof. Luc Bouganim and my co-advisor Prof. Benjamin Nguyen for the continuous support of my Ph.D. study and related research. I would like to thank them for their guidance, help and support that they have given me all these years. I am very grateful for their patience, motivation, enthusiasm, and immense knowledge, as well as their criticisms, their human qualities and encouragement that contributed to the success of this thesis.

Besides my advisors, I would like to express my sincere thanks to the rest of the thesis committee: Prof. Philippe Bonnet, Prof. Sébastien Gambs, Prof. Jean Marc Petit, and Matthieu Manant who were willing to dedicate part of their time to my Thesis. Particularly, I appreciate the precise and efficient reviewing of the complete thesis they performed and I would like to warmly thank them for the insights and the comments they gave me. I am privileged and honored they accepted to be committee members of this thesis.

My sincere thanks also go to Prof. Philippe Pucheral who provided me an opportunity to join SMIS team as intern, and who gave me access to the laboratory and research facilities. Without his precious support it would not have been possible to conduct this research. Moreover I would like to thank Matthieu Manant, Nicolas Soulié, Fabrice Le Guel, Grazia Cecere and Vincent Lefrere for fruitful discussions and valuable suggestions, which helped in conducting some survey experimentations.

I thank also all the members of the SMIS team that allowed me to make my dream come true by giving me the necessary support in order to successfully finish my PhD Thesis in a pleasant work environment. Special thanks to my fellow lab mates in SMIS, Quentin Lefebvre, Aydogan Ersoz, Quoc-Cuong To, Saliha Lallali, Paul Tran-Van, Julien Loudet for the stimulating discussions, for the times that we were working together before deadlines, and for all the fun we have had in the last years. I would also like to thank my lab mates Riad Ladjel, Razvan Nitu, Soukayna Lafteh, Maggie Mache Mbongtot and Rémy Pasquion who joined recently SMIS team for the interesting conversations that we had.

Last but not the least; I would like to thank my family: my parents Konstantinos and Alexandra, as well as my sister Maria-Rafailia for supporting me spiritually throughout my life. Finally, I would like to thank my friends and people that are close to me for their company and support. I dedicate this work to my family and to the whole SMIS-team.

Table of contents

List of figures

List of tables

Chapter I
Introduction

In this chapter, we first position the context of this study; then we list its precise objectives, present the main contributions and give the outline of this thesis.

1.1 Context of the Study

The XXIth century has seen the advent of online social networks (OSN), the best known being obviously Facebook. In social networks, users (individuals) share information about themselves: pictures, videos, text messages, opinion polls, etc. The process of information sharing and spreading depends on the social network, but in general, a user publishes information, visible by her friends, and these friends can then decide to republish the information.

This information is most often loosely structured: some information, that we will call data, is of a specific type, such as a picture or photograph. This data is enhanced with metadata, such as location of the photo, date/time, or the list of individuals who were in the picture. Metadata can be automatically generated (e.g. by the camera when the picture is taken), or constructed manually and collaboratively by the users themselves. A typical example is tagging one's friends in a photo. Also note that most of this data is personal data, in the sense that it pertains to individuals. In Europe, the processing of such data falls in the scope of data protection regulations[1]. Moreover, there is a general consensus that this data should be processed with great care, and that more control should be returned to the data owner, especially since this data is going to be shared with many other users. In this thesis, we focus on one of the aspects of control, i.e. usage control, which we will define next, and detail in Section 2.1.

In short, Usage Control (UCON) is a mean to specify how a user or producer of data can control the usages of this data. UCON does not only manage access to data, but

[1] http://www.allenovery.com/SiteCollectionDocuments/Radical%20changes%20to%20European%20data%20protection%20legislation.pdf

also what can be done with this data. UCON is paramount in the context of OSN, where users wish to share their data, but also would like to keep control over it, especially since it will end up disseminated in many different places, and in many different hands. However, although the theory of UCON has already been discussed [Park02, Park04], a fully-fledged system built around this concept has yet to be studied.

1.2 Addressed Issues

The central issue addressed in this thesis is that of data sharing, and the subsequent usage control issues. The core question studied is how can a user control how their data is shared? The results are applied to two different classes of applications: Online Social Networks and Online Surveys. In both of these applications, as in many others, the general consensus is that controlling how one's data is shared is of utmost importance. However, there are currently very few systems that can be used to realize this end.

It is important to note that this thesis was driven by a multi-disciplinary goal in the context of the Digital Society Institute (Institut de la Société Numérique, ISN) of Paris-Saclay and its Privacy working group, which regrouped jurists, economists and computer scientists. Thus, one of the specificities of this thesis is to seek to address the issue of UCON through two axes: (a) computer science and (b) economics.

The computer science axis covers algorithms, models and system design, and the socio-economic axis covers field-testing with real users, to measure usability and adaptability of both the concepts and systems designed. Indeed, during this Ph.D. we had strong collaborations with researchers in field economy from the Réseaux, Territoires, Innovations, Mondialisation lab of University of Paris-Sud (RITM). This collaboration resulted in a large-scale pre-experiment presented in Section 4, with exciting perspectives.

In consequence, whenever a concept or application is proposed, we always kept in mind how this concept would translate "on the field", and were interested in studying how it would be welcomed by users.

1.3 Contributions

This thesis makes three contributions to the field of privacy:

(1) a prototype implementation of the UCON model, applied to an OSN scenario,
(2) an algebraic extension to UCON to control the complex sharing of data, using XQuery operators,
(3) the design, implementation and field testing of two secure platforms: the first one being used in privacy lab courses in order to help students design Privacy-by-Design applications and the second one to manage sensitive data collected through online forms.

Having studied some traditional and modern access control models, we realized that these models are not sufficient to cover the needs of modern environments due to some limitations. For instance, these models focus on authorization only before the access is allowed and there is no ongoing control concept to be taken into consideration. In this direction, we have studied $UCON_{ABC}$, a powerful usage control model that can answer to these insufficiencies. Thus, in the first contribution of this thesis, we study how it is possible to implement UCON in an environment that we assume is trusted.

UCON can answer correctly the question of sharing an object with complex rules with identified users. However, in the context of OSNs, the problem is that we may want to share many data with many users, and user's circles, but not necessarily exactly the same version of data. Then, we have decided to extend UCON implementation such that it will be able to control the simultaneous sharing of data to several circles, with different data precision granularities. Thus, we proposed our second contribution; called DatShA. DatShA is a data sharing algebra. It builds upon the concept of UCON, and enables a user to specify privacy-preserving workflows on their data. One originality of this approach is that these workflows can be combined with other workflows, or executed by other users on their own data. The objective of this work is to give non-expert users a mean to exert usage control through example, by reusing algebraic plans proposed by expert users.

Our final contribution is multidisciplinary. It is divided in two parts. The first part includes a Secure Portable Token (SPT) based Data Sharing platform. Our approach provides the students with a simple platform (minimal basic blocks) such that they are able to build their own modules or Privacy-by-design applications on top of it. For instance, the students are able to implement the concepts presented in UCON and DatShA, using this platform. It is worth mentioning that it was used by around one hundred and fifty students, at ENSIEE in 2015 and at ENSIEE, UVSQ, and INSA in 2016. The second part consists of the design and implementation of an online form based survey platform, built using secure hardware and trusted software (the PlugDB server running on the SPT [Allard10, Anciaux10, Anciaux13, Anciaux15a, Anciaux15b]). This system has been tested in the context of the PAIP[2] and Valdo[3] projects on over one hundred and forty non expert users. This platform will be reused by economists of RITM through a “mobile experimental lab”, in order to gather personal data for research purposes, while protecting the privacy of users. The link between two platforms described is that both aim to face the technology with users.

1.4 Outline

This thesis is composed of two main parts. The first part (Chapter 2) presents the background knowledge necessary to understand the approaches proposed and positions it with respect to related work.

The second part is composed of Chapter 3 and Chapter 4. Chapter 3 details the design and implementation of the UCON model, presenting two approaches. Both approaches have been implemented in Java; the former combined with XML and the latter with SQL. Moreover, based on UCON, we present a data sharing algebra that allows users to specify privacy-preserving workflows on their data. Chapter 4 concentrates on the experimental work that has been done during the Thesis. More specifically, the first part of this chapter focuses on the implementation of a data sharing platform that is currently used in the context of privacy lab courses. This platform allows users to share their

[2]https://project.inria.fr/smis/peps-paip-pour-une-approche-interdisciplinaire-de-la-privacy-digital-society-institute-dsi-sept-2013-sept-2014/

[3]http://www.cerdi.u-psud.fr/partenariats/projet/valdo-valorisation-et-monetisation-des-donnees-personnelles-a-lere-du-big-data

content under restrictions (i.e. access and usage control policies) as well as to develop and extend this platform in order to be more powerful, in terms of security aspects. The second part of this Chapter is dedicated to a pre-experimentation that has been conducted, related to sensitive questionnaire surveys. In particular, two approaches - a secure and a vulnerable one - of the same system are described in order to point out the importance of sensitive information protection.

Finally, Chapter 5 concludes and proposes some ideas for future works.

Chapter II
Background Knowledge and Related Works

This Chapter provides the necessary background knowledge in order to understand the contributions of this thesis. As we referred in Chapter 1, the main objective of this thesis is to cover issues relating to data protection and sharing, providing solutions that address the subsequent access and usage control issues that may occur. Thus, in the first two Sections, the related works address some important access and usage control models that motivated this work. Particularly, we start by introducing some well-known Access Control Models, such as the Discretionary Access Control (DAC), the Mandatory Access Control (MAC) and the Role-Based Access Control (R-BAC) Model. Then, having noticed the difficulty of traditional Access Control models in covering the needs of modern information systems due to lack of dynamic authorization controls (i.e. ongoing control, obligation-based and condition-based controls), lack of policies' enforcement during the access and lack of rights' definition, we give and analyze the background knowledge required for understanding the field of usage control, discussing some well-known existing models that could respond to these insufficiencies. We describe $UCON_{ABC}$, a usage control model that we selected given its powerful features such as mutability of attributes and continuity of decision access. These features motivated some implementations around $UCON_{ABC}$ (see Chapter 3). We also explore the domain of OSN presenting the sharing policies and techniques that are used, as well as existing systems that allow users to apply fine-grained controls to their personal data. Finally, we overview some existing varieties of personal data servers that contributed or can potentially contribute to this work.

2.1 Access Control Models

Access control is the process of mediating each request to resources and data maintained by a system and determining whether the request should be granted or denied [Samarati01]. An Access Control model defines policies and provides descriptions of the security properties of a system. The entities of this model include subjects (e.g. user, process) and objects that contain the information that the subjects would desire to access.

An access control policy defines the rules that regulate access to resources determining which user is allowed to access which objects. A system is considered secure when the access control mechanism implements an access control model and when we are able to prove its security.

2.1.1 Traditional Access Control Models

Access control models are often classified as either discretionary or non-discretionary. The three best-known models include the Discretionary Access Control model (DAC), Mandatory Access Control model (MAC), and Role-Based Access Control model (R-BAC).

2.1.1.1 Discretionary Access Control

Discretionary access control [Li11,Mao11] defines some authorization rules, using user's identity in order to determine the user’s privileges within a system. The owner of resources determines the policy which is related to the decision of which users are allowed to access which object, and what kind of access, including the privileges that they will have [NCSCUS87]. The discretionary model is represented by an Access Control Matrix (ACM) [Lampson74], which is either a structure or a table where a subject can be represented on each row and an object can be represented on a column. The entries of the table's cells represent the rights that a subject can exercise on an object.

Two important concepts can be identified in DAC [Li11,Mao11]:

- **File and data ownership**
  Every object in the system has an owner. In most cases in a DAC system, the initial owner of an object is the subject that caused it to be created. As far as the access policy for an object is concerned, it is determined by its owner.

- **Access rights and permissions**
  Access rights and permissions constitute the controls that an owner of the objects can assign to other subjects, for particular resources. As far as access controls are concerned, they may be discretionary, listed in an Access Control List (ACL).

One of the major weaknesses of DAC is that programs inherit the identity of the invoking user. This weakness makes the system vulnerable to malicious programs. There is no real security on the flow of information in such a system.

Another weakness of DAC lies in the fact that only the owner can determine which object could be accessed or not by a given subject. There are no constraints that are related to the usage of information that the user has received. Finally, the owner could not be able to automatically revoke access, since there are no specific constraints (i.e. temporal, spatial) to be used to restrain the usage of the object.

Unix, Linux, Windows access control is based on DAC. As another example, capability systems[1] provide discretionary controls since they permit subjects to transfer their access to other subjects.

[1] systems that use protected object references as their fundamental security primitive.

2.1.2.2 Mandatory Access Control

In Mandatory Access Control [Denning76, Hu11, Shan12, Osborn97] there is a labeling mechanism which is employed in order to label an object in a system. Thus, MAC is able to control the information flow between different objects within a system. For instance, the subjects and objects that belong to a system should have labels that are assigned to them. The sensitivity label of the subject and object specifies their level of trust/sensitiveness. A security policy is specified in order to determine the information flow of objects between subjects which have specific labels and can be enforced according to some specific security requirements of an organization.

Two methods [Hu11, Osborn97] are used for applying mandatory access control:

- **Rule-based access control**
  This type of control defines some specific conditions for accessing the requested object(s). A Mandatory Access Control system implements a simple form of rule-based access control in order to determine if the access should be either granted or denied by comparing the object's and subject's labels.

- **Lattice-based access control**
  A lattice-based access control model (LBAC) is a complex model that is based on the interaction between subjects and multiple objects. A lattice defines the levels of security that an object may have and that a subject may have access to. Indeed, the security level access can be expressed in terms of the lattice (e.g. a partial order set P) where each object and subject have a greatest lower-bound (meet) or least upper-bound (join)[2] of access rights. Hence, the subject is allowed to access an object only if the security level of the subject is greater than or equal to the security level of the object.

[2] If a and b are elements from P, the join is denoted as $a \vee b$ and the meet is denoted $a \wedge b$

2.1.2.3 Role-Based Access Control

Role-based access control (R-BAC) [Ferraiolo03, Beresnevichiene03, Sandhu96a, Sandhu96b] is an access policy which simplifies the specification of authorization rules by grouping the users of a system into roles, according to their duties and authorizations within the organization.

A role in R-BAC can be viewed as a set of permissions. There are three primary rules [Sandhu96] that are defined for R-BAC:

- **Role assignment**
  If the subject has selected or been assigned a role, then s/he can execute a transaction.

- **Role authorization**
  If the subject has selected or been assigned a role and the subject's active role is authorized for the subject, then the user can only take on the roles for which s/he is authorized.

- **Transaction authorization**
  A subject can execute a transaction if the transaction is authorized for the subject's active role. In the previous rule, if the subject has also selected or been assigned a role and the subject's active role is authorized then this rule ensures that the user can execute only transactions for which s/he is authorized.

R-BAC can improve significantly the administration and management of the permissions which are assigned to the users (i.e. permissions are given to roles). The members who belong to a role inherit the permissions that are assigned to the role and the decision for access is based on the role of a user. The permissions which are associated with a role can also be changed according to the changes in the organizational structure.

R-BAC is a non-discretionary model, such as MAC, but it handles the permissions differently. MAC controls I/O permissions based on a user's clearance level and additional labels. In R-BAC collections of permissions that include complex operations such as an e-commerce transaction are controlled. R-BAC differs also from DAC, since

DAC allows users to control access to their resources, in contrast to R-BAC, where the access is controlled at system level, beyond the user's control.

According to Sandhu [Sandhu96a], R-BAC could be used to implement traditional multilevel security policies. R-BAC examples include commercial applications and also military systems. Systems including Microsoft Active Directory, Microsoft SQL Server, SELinux, grsecurity, FreeBSD, Solaris, Oracle DBMS, PostgreSQL 8.1, implement some form of R-BAC.

2.1.2 Modern Access Control and Digital Rights Management

Research in the Access Control domain has shown that access control needs enhancements in order to meet the needs of modern applications and systems. New concepts which include Trust Management and Digital Rights Management (DRM) have emerged. In this section, we are going to discuss these concepts.

2.1.2.1 Trust Management

Authentications and authorizations of users, computers, electronic devices and networks that rely on user authentication and access control are widely used. This is insufficient for open environments, such as the Internet, since as an open environment, it requires flexible and dynamic access control. In addition, distributed systems lack central control and also its users cannot be predetermined by any means. Trust management was developed in order to fill the gaps and enhance the traditional access control.

In a distributed computing environment, the amount of people who request a resource is quite large. In the majority of web services it is common that the subjects are unknown to the system prior to the access request. In Trust Management (TM) [Blaze96], there is no need of a predefined identity in order to authorize the subject. The authorizations are based on the credentials of the subject rather than a predefined identity, such as a username or/and a password. Trust-management engines do not need to resolve

“identities” in an authorization decision. Instead, these engines express privileges and restrictions in a programming language, providing in that way increased flexibility, opening the way to modern, scalable security mechanisms.

Trust Management systems include systems that have focused on authentication [Wobber94], general-purpose authorizations [Blaze96, Ellison99], and those that were developed to cover specific needs [Balfanz00, Herzberg00].

2.1.2.2 Digital Right Management

Digital Rights Management (DRM) [Guth03, Liu03, Rump03, Gooch03, Paskin03] addresses the information security that is needed to businesses dealing with digital content (i.e. books, music, images, and videos).

DRM allows restricting the access to copyrighted digital content [Guth03, Liu03] which is being distributed to the clients. The usage of the object is controlled by a client side reference monitor. This monitor is responsible for protecting against copyright violation and also allows only legitimate users to use the copyrighted material according to some specific conditions which are defined by its owner.

DRM systems' examples [Guth03] include those that have been developed by various organizations for commercial purposes and are concerned with the protection of intellectual property rights. For instance, it is worth mentioning Fair Play by Apple, Windows media DRM by Microsoft and Adobe's protected streaming. However, there is no compatibility between the aforementioned examples. This issue complicates and affects the integration of DRM solutions into some mainstream digital devices that are used by individuals who are the consumers of digital material.

2.2 Usage Control Models

The notion of usage control includes the integration of obligations or/and conditions in existing access control models. In this Section, we are going to present some usage

control models that contributed to this thesis. Moreover, we present in details how UCON model specifications can be expressed in XACML[3], according to the literature.

2.2.1 $UCON_{ABC}$ model

In this Section, we are going to describe in details one of the well-known usage control models, named $UCON_{ABC}$ [Park02, Park04]. The $UCON_{ABC}$ model consists of eight core components that are involved in the authorization process (see Figure II - 1), divided into main components, decision factors and decision properties. We are also going to present some scenarios to point out its power. Two implementations of this conceptual model are going to be presented in Chapter 3.

2.2.1.1 The UCON Model and the Reverse UCON model

Main components of $UCON_{ABC}$ comprise:

- **Subjects:** Entities which are associated with attributes. They hold and exercise certain rights on objects (i.e. user, group, role, or a process). These entities could be either consumer subjects (CS), which receives rights and objects and use those rights in order to access the objects (i.e. e-book reader), provider subjects (PS), which provides an object and hold certain rights on it (i.e. author of an e-book), or identifiee subjects (IS), who are identified in digital objects, including privacy-sensitive information (i.e. patient of a health care system).
- **Subject attributes**: Properties of the subjects that can be used for the authorization process (i.e. identities, roles, credits).
- **Objects:** Entities which are associated with attributes. These entities could be either privacy sensitive or privacy non-sensitive and original or derivative. In the former case, privacy sensitive objects include individually identifiable information

[3] eXtensible Access Control Markup Language: a general-purpose access control policy language (https://www.oasis-open.org/committees/download.php/2713/Brief_Introduction_to_XACML.html).

that causes privacy problems if it is not used in an appropriate way. In the latter case, a derivative object means derived or cited from an original work in order to create another digital work.

- **Objects attributes**: Certain properties that can be used for the authorization process (i.e. security levels, ownerships).
- **Rights:** Privileges that a subject could hold on an object. These privileges consist of a set of usage functions which enable the access of a specific subject for specific objects (i.e. watching a movie). Rights can be divided into consumer rights (CR), provider rights (PR), and identifiee rights (IR).

Decision Factors comprise:

- **Authorizations**: Functional predicates that should be evaluated for a usage decision. These predicates evaluate subject attributes, object attributes and requested rights together with a set of authorization rules for the usage decision. Authorizations could be either pre-authorizations (preA) which are performed before a requested right is exercised or ongoing-authorizations (onA) which are performed while the right is exercised.
- **Obligations**: Functional predicates, verifying mandatory requirements that a subject should perform before or during a usage. Obligations could be either pre-obligations (preB) which can utilize some kind of history functions in order to check whether some specific activities have been fulfilled or not or ongoing-obligations (onB) which should be satisfied continuously or periodically while the allowed rights are in use.
- **Conditions:** Set of decision factors that the system should verify during the authorization process along with authorizations before allowing usage of rights on a digital object. Condition predicates evaluate current environmental status (i.e. accessible time, location) in order to check if relevant requirements are met or not. These predicates cannot be mutable, since conditions are not under direct control of individual subjects. Evaluation of conditions cannot update any subject or object attributes.

Decision Properties include:

- **Mutability of attributes**: Feature of $UCON_{ABC}$ that provides the model with flexibility in order to accommodate complex access control scenarios that are met in modern computing environments, including the values' modification (i.e. subject, object or environment attributes) as a result of access to an object. For instance, let us assume Alice who would like to read an e-book. She should pay in order to access the digital content of the e-book. Once she accesses this e-book, then her account balance is reduced. Then, the permissions of Alice (identifiee subject) for the e-book (object) and the state of the system (i.e. access is allowed, balance reduced) are affected.

- **Continuity of decision access**: Feature of $UCON_{ABC}$ enabling the evaluation of authorizations and conditions not only before access but also during the usage of the object (when a subject is exercising the rights and permissions they are granted to subject on an object). For instance, the provider of a movie restricts its access to 70 seconds only, for advertising purposes. If a user would like to watch the whole movie, a subscription fee should be paid by him, otherwise only a sample video will be available. The system will keep track of the time and revoke its access as soon as the time duration of 70 seconds expires.

By obtaining or exercising usage rights on a digital object, such as on an image, mp3 or a video file, another digital information object may be created. This new object is called derivative object. It includes privacy-related information and needs controls for its access and usage, as for the original object. Thus, usage control is reversed, since the provider subject now becomes the consumer subject.

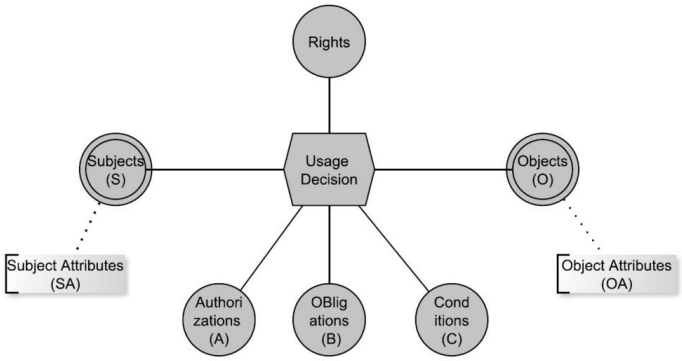

Figure II - 1. $UCON_{ABC}$ Model Components

Let us consider Alice, who is the consumer subject (CS), who would like to watch an *.avi movie (see Figure II - 2). In order to obtain watch rights, she should agree on payment-per-watch, which is an obligation (OB). She should provide information of her credit card. When she exercises the watch rights, she has to report her usage log on the video, which is also an obligation (OB). Alice is both a provider subject (PS) and identifiee subject (IS) of the log/payment information and may hold certain rights (PR and IR) on them (i.e. delete her ID from log), after the payment. Both payment information and log information constitute derivative objects. The distributor could be a video production company that may have rights to collect log information either by adding an obligation on consumer rights or giving consumer rights to get some store credits on log reports. If Alice has rights to get some store credit based on her watch time, then it is a distributor obligation to issue these credits to her.

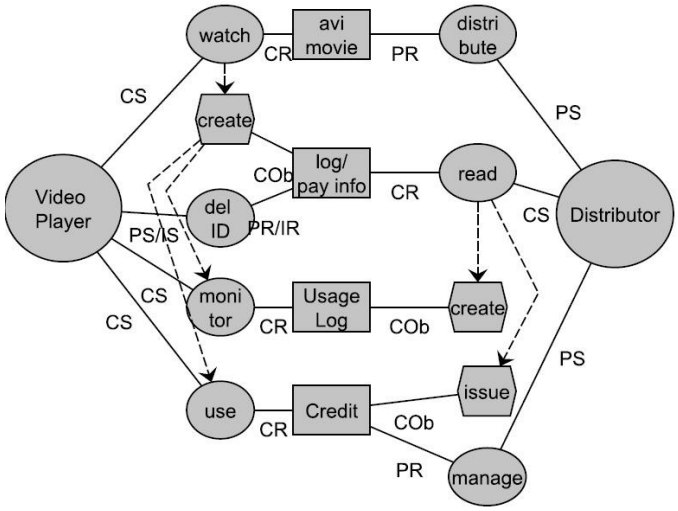

Figure II - 2.Reverse UCON model

2.2.1.2 $UCON_{ABC}$ Core Models

R. Sandhu and J. Park [Park02, Park04] have developed a family of core models for usage control, based on three decision factors: authorizations, obligations, and conditions, along with continuity of access decision and mutability of attributes in order to support modern access control requirements, enhancing the domain of access control. These models have focused on the enforcement process ("core" models), excluding any administrative issues.

If all attributes are immutable, no updates are possible as a consequence of the decision process. Section 2.1.1 discussed that mutability of attributes allows certain updates either on subject or object attributes as side effects of usages. Thus, for mutable usage, updates are required either before (pre), during (ongoing), or after (post) the usage. Based on these criteria, they have developed 16 possible model spaces for usage control (all possible combinations).

Figure II - 3.a,Figure II - 3.b,Figure II - 3.c, shows the three basic models: $UCON_A$, $UCON_B$ and $UCON_C$, accordingly while Figure II - 4presents some possible combinations of $UCON_{ABC}$ models.

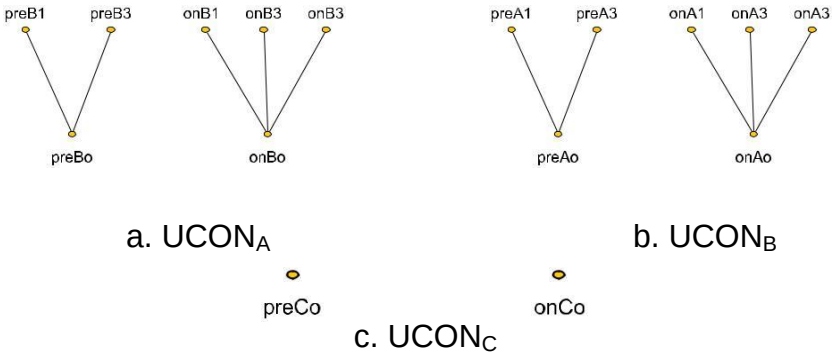

Figure II - 3. Basic Models

UCON$_A$ Models

- **UCON pre-Authorizations Models**

  In UCON preA models [Park02, Park04], the authorization decision process is done before usage is allowed. PreA examines usage requests using subject attributes, object attributes, and rights and then decides whether the request is allowed or not. Three detailed models exist based on mutability variations. Pre-updates and post-updates on subject and object attributes are optional procedures to perform update operations on them (see Figure II –3.a)

- **UCON ongoing-Authorizations Models**

  In UCON onA models [Park02, Park04], usage requests are allowed without any 'pre' decision-making and authorization decisions are made continuously or repeatedly while usage rights are exercised. Currently allowed usage right is revoked when a certain requirements become dissatisfied, as a result its exercise is stopped. Four detailed models exist (see Figure II –3.a).

UCON$_B$ Models

- **UCON pre-Obligation Models**

In UCON preB models [Park02, Park04], pre-obligations should be fulfilled before access is permitted. PreB is a kind of history function that checks whether certain obligations have been fulfilled or not and returns true or false for the usage decision (see Figure II –3.b).

- **UCON ongoing-OBligations Models**

  In UCON onB models [Park02, Park04], usage requests are allowed without any 'pre' decision-making. By ongoing authorizations, monitoring is actively involved in usage decisions while a requested right is exercised (see Figure II –3.b).

$UCON_C$ Models

Generally, $UCON_C$ models cannot be mutable. However, the value of conditional status can be changed as the environmental situation is being changed (i.e. current time is changed as time goes; a wireless access point is changed as a user moves around a building). Subject or object attributes are not used for usage decision process but they are used for taking a decision related to what kind of condition elements (preCON) have to be enforced for usage decision.

- **UCON pre-Conditions Models**

  In UCON preC, conditions constitute environmental restrictions that should be satisfied for usages. Generally, preCON are environmental restrictions that are not related to subjects and objects (see Figure II –3.c).

- **UCON ongoing-Conditions Models**

  Environmental restrictions have to be satisfied while rights are in active use. This could be supported within UCON onC model [Park02, Park04]. In this model, usages are allowed without any decision process at the time of requests and there is an ongoing conditions predicate in order to check certain environmental status repeatedly throughout the usages, as well (see Figure II –3.c).

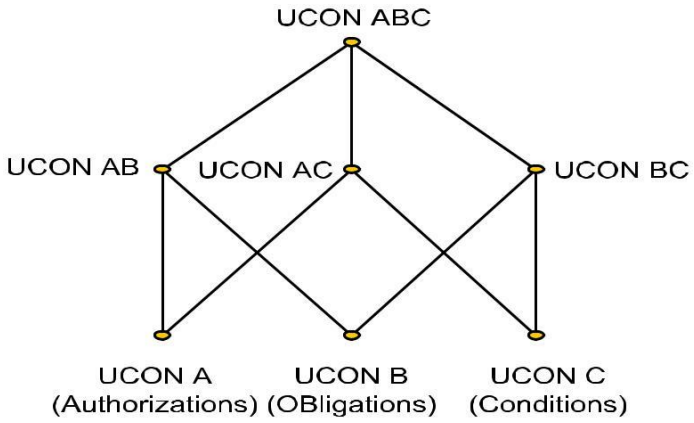

Figure II -4. $UCON_{ABC}$ Family of Core Models

2.2.2 Policy Specification of Usage Control

XACML [XACMLIntro] is a policy language promoted by the OASIS consortium. It introduces new data types, identifiers, elements and functions; hence it offers a generic policy structure and interoperability feature that is able to represent the features of the UCON model and is adaptable to various platforms and environments.

The UCON model is able to cover a wide range of applications, including social networking, health care systems and digital rights management (DRM), as described in Section 2.1. Thus, the formal specification of models should be expressed in a generic policy language like XACML, since it provides identifiers for subject categories (e.g. access-subject, recipient-subject, intermediary subject), attribute categories (e.g. action, resource), rules and expressions (e.g. rules, obligations, conditions).

According to the literature [XACMLIntro] the definition of a subject includes the subject-type identifier along with the XACML subject category in order to specify the type of accessing subject.

```
<xs:element name="SubjectAttributeDesignator"
type="xacml:SubjectAttributeDesignatorType"
substitutionGroup="xacml:Expression"/>
<xs:complexType name="SubjectAttributeDesignatorType">
<xs:complexContent>
<xs:extension base="xacml:AttributeDesignatorType">
<xs:attribute name="SubjectCategory" type="xs:anyURI" use="optional"
default="urn:oasis:names:tc:xacml:1.0:subject-category:access-subject"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
```

Similarly, XACML depicts that objects can be defined using resource category as follows:

```
<xs:element name="Resource" type="xacml-context:ResourceType"/>
<xs:complexType name="ResourceType">
<xs:sequence>
<xs:element ref="xacml-context:ResourceContent" minOccurs="0"/>
<xs:elementref="xacml-context:Attribute" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
```

Regarding actions, OASIS proposed the action-type identifier that can be used to represent action categories of UCON.

```
<xs:element name="Action" type="xacml-context:ActionType"/>
<xs:complexType name="ActionType">
<xs:sequence>
<xs:element ref="xacml-context:Attribute" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
```

Authorizations can be defined in XACML as general rules. The rule-id attribute includes the time of rules evaluation (i.e. pre-authorization or ongoing-authorization).

```
<xs:element name="Rule" type="xacml:RuleType"/>
<xs:complexType name="RuleType">
<xs:sequence>
<xs:elementref="xacml:Description" minOccurs="0"/>
<xs:element ref="xacml:Target" minOccurs="0"/>
<xs:element ref="xacml:Condition" minOccurs="0"/>
</xs:sequence>
<xs:attribute name="RuleId" type="xs:string" use="required"/>
<xs:attribute name="Effect" type="xacml:EffectType" use="required"/>
</xs:complexType>
```

In order to define mandatory actions that the subject is able to perform, XACML provides the obligation expression element.

```
<xs:element name="Obligations" type="xacml:ObligationsType"/>
<xs:complexType name="ObligationsType">
<xs:sequence>
<xs:element ref="xacml:Obligation" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
```

Finally, OASIS proposed the condition element that contains single expression elements (e.g. functions to be evaluated). The evaluation-phase attribute with the values pre-access or ongoing access specifies pre or ongoing conditions, accordingly.

```
<xs:element name="Condition" type="xacml:ConditionType"/>
<xs:complexType name="ConditionType">
<xs:sequence>
<xs:element ref="xacml:Expression"/>
</xs:sequence>
</xs:complexType>
```

2.2.3 A Usage Control Based Architecture for Cloud Environments

Carniani et al. [Carniani16] propose a framework that includes advanced authorization services in order to control the resources in the Cloud. The design of the framework is based on the $UCON_{ABC}$ model, introducing the U-XACML language that extends XACML, simplifying the definition and the management of access and usage control policies in a distributed environment.

Some enhancement that U-XACML language provides regarding the definition of policies include the DecisionTime attribute with values pre (pre-decisions) and on (on-decisions) in the <Condition> element to specify when the evaluation of this condition is going to be executed. Similarly, U-XACML extends <ObligationExpression> element by adding the DecisionTime attribute, as well, such that its values pre (pre-obligations), on (on-obligations) and post (post-obligations) to define the obligation evaluation time. Mutability of attributes is represented in U-XACML by <AttrUpdates> element and contains also the attribute <UpdateTime> with the values pre (pre-update), on (on-update), and post (post-update) in order to specify when update actions have to be executed.

The most powerful feature that this framework adopts from the conceptual usage control model is that the policies are continuously enforced by this framework, taking into consideration factors that may change over time (mutability of attribute values), as well as the duration of users' actions (ongoing accesses).

Figure II -5 shows the framework that they propose. It consists of two main components named Cloud provider (on the left) and Usage Control service (on the right). Moreover, this architecture includes three horizontal layers:

- **Application Layer**: Cloud IaaS[4] service, as well as other services that are needed for policy evaluation, regarding attributes management.
- **Usage Control Layer**: Usage control system's core components. Regarding Cloud services, it refers to a Policy Enforcement Point (PEP), while for Usage Control service it is represented by the part that keeps the components that are needed for the ongoing evaluation of the policies.
- **Communication Layer**: It performs the communication between Cloud provider and the Usage Control service.

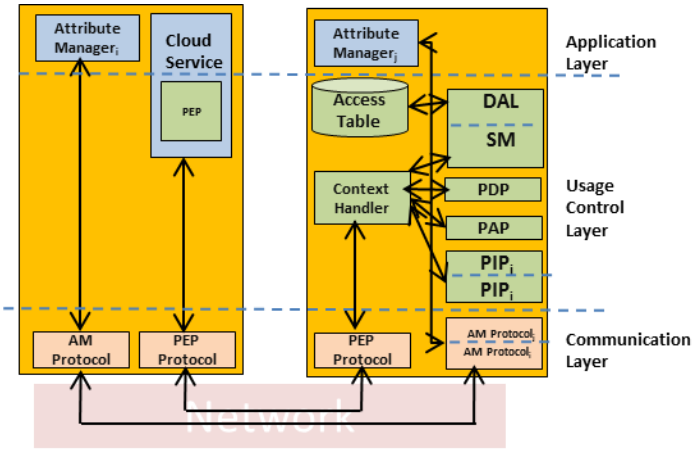

Figure II -5. UCON Architecture for Cloud Environments

Considering the concepts described, we can talk, for instance, about a cloud-based usage control engine on top of personal cloud (e.g. CozyCloud) able to provide stronger protection of personal sensitive data.

[4] https://www.profitbricks.com/what-is-iaas

2.2.4 UseCON: A Use-Based Usage Control model

Grompanopoulos et al. [Grompanopoulos12] having noticed $UCON_{ABC}$ model's limitations (e.g. no historical information regarding denied requests or revoked usages is recorded, hence attribute mutability based on that information is not supported), propose a Use-based Usage Control model (UseCON) that constitutes an extension of UCON model, since it provides an enhanced utilization of the usage decision criteria and also an enhanced support for complex usage modes, taking into account current usages exercised in the system.

The main components of this model include subject, object, action and use (see Figure II - 6), each of it associated with attributes.

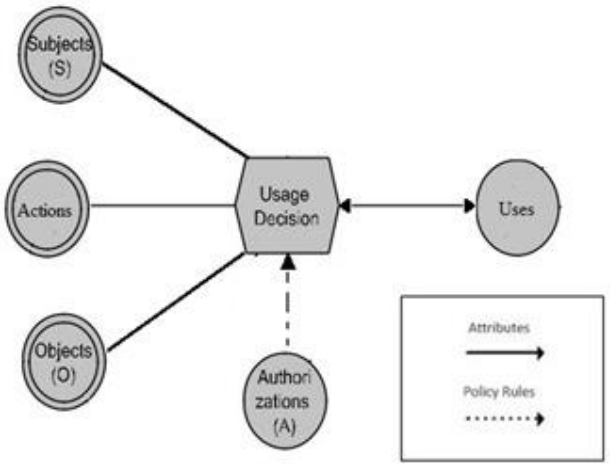

Figure II -6. UseCON elements and relations

Similarly to UCON, subjects are defined as entities that are able to exercise operations on objects, while objects can include either entities or services. Attributes are defined as

security-relevant properties or capabilities[5]. Actions include operations that the subjects can perform on objects. Actions are also associated with attributes.

In this model, only authorizations are used to control the usage of resources. Pre-authorizations and ongoing-authorizations are supported as in UCON.

However, the main component of the UseCON model is the component "*use*" that includes the appropriate information in order to take the decision regarding resources access allowance. This information is created when the access of specific objects is requested and records the relation between these components (i.e. pair of subject, object and action). Uses are associated with attributes as well. These attributes include information that is related to any pair of these components (subject, object, action). Moreover, uses are associated with a state attribute that represents all the possible states of a use (see Figure II - 7). These states include:

- **Requested State**: A usage request has been done.
- **Activated State**: A requested usage is allowed, since pre-authorization rules have been satisfied.
- **Denied State**: A requested usage has been denied because the pre-authorization rules have not been satisfied.
- **Stopped State**: An allowed usage has been terminated by the system because the on-going authorization rule is not satisfied anymore.
- **Completed State**: A usage has completed due to subject's action.

[5] A capability is a token, ticket, or key that gives the possessor permission to access an entity or object in a computer system (e.g. capabilities for file access, capabilities for memory access, etc.) - Dennis and Van Horn, 1966.

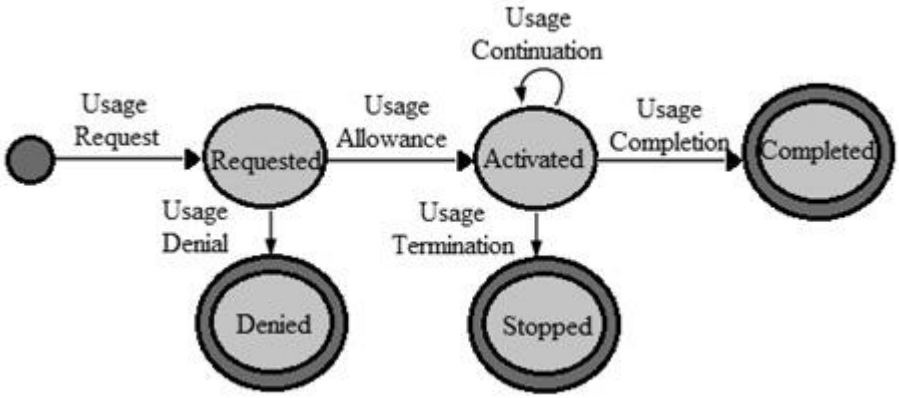

Figure II -7. Use state-transition diagram

2.2.5 SoNeUCON$_{ABC}$: A usage control model for Web-Based Social Networks

In [Manzano13, Manzano14] Manzano et al. identify a set of requirements such as relationship management, privacy-preserving and fine-grained access control management, interoperability, sticky-policies[6], and data exposure minimization[7] in order to provide an expressive usage control model for an OSN. Their work is focused on the definition and implementation of the SoNeUCON$_{ABC}$ model. This model extends UCON$_{ABC}$ such that it can cover exhaustive relationship management. Relationships can be entity attributes of the relationship. For example, Alice has a list of 'friends', 'colleagues', and 'relatives'. These lists could be an example of such user's attributes.

[6]Conditions and constraints attached to data that describe how data should be treated, preventing from uncontrolled data disclosures after being released (related to usage control) (see https://documents.epfl.ch/users/a/ay/ayday/www/mini_project/Sticky%20Policies.pdf).

[7]A mechanism to deal with undesirable accesses to OSN users' data, as well as to conceal data from servers, protecting users from privacy violations and also minimizes unauthorized data exposures.

The main components of the SoNeUCON$_{ABC}$ model (see Figure II – 8) include subjects, objects, relationships (i.e. relations between a pair of users) associated with attributes, rights that refers to actions over objects, authorizations that corresponds to rules that have to be satisfied so that subjects may access objects, obligations that refer to requirements that have to be satisfied before or during the usage process and Conditions that are requirements regarding context features.

Our work can benefit from the use component that the UseCON model offers, along with the relationship component of this model. Having combined the aforementioned components, we will have a pair of subject, object and actions available along with the relations between a pair of subjects, hence we will be able to define and illustrate more complex scenarios in the context of OSN.

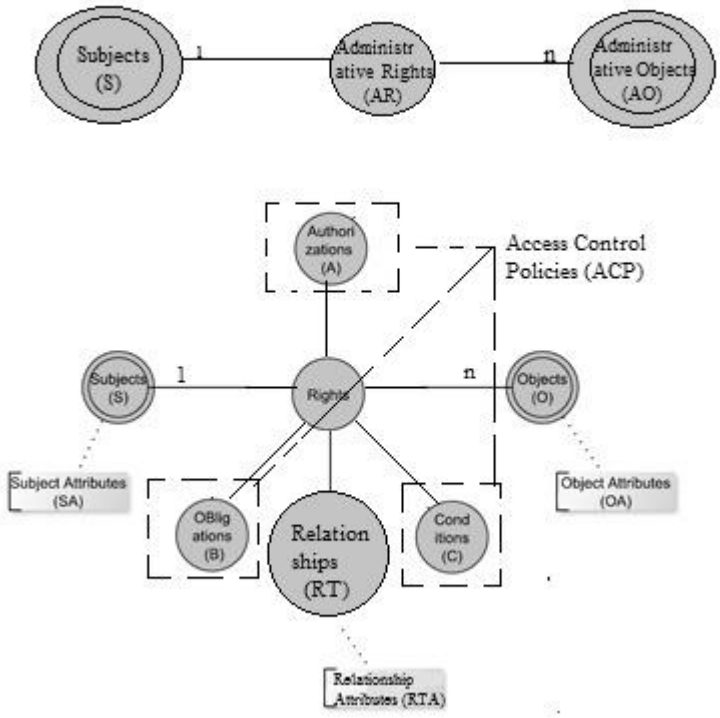

Figure II -8. SoNeUCON$_{ABC}$ model

2.3 Sharing Policies in Social Media

Online social networks (OSNs) are used by hundreds of millions of individuals for personal information and digital resources sharing. Thus, there is a need for functionalities such that they will be able to manage their published information securely. Such functionalities include access and usage control policies that can be specified by the resources owners. In this section, first we present the notion of Online Social Networks, pointing out their weaknesses, second we discuss privacy nudges[8] for Facebook and third we describe Penumbra, a tag-based, logical access-control framework for personal file sharing.

2.3.1 Online Social Networks

Online social networking websites[9] are online platforms that allow users to connect and interact with other users via their public profile. These websites include users who create a public profile and then create a list of users with whom they share a connection. However, once the request has been sent, OSN allow the users on the list to confirm or deny the connection. Once the connections are established, then the users can interact diversely (e.g. exchange instant messages, photos, audio or video).

Recently, OSN have become very popular and thus they have attracted the attention of many research communities who contribute to the existing literature on OSN ([Milgram77, Watts03]), as well as the media ([Arrison04, Black04, Leonard04, Newitz03, Sege05]). OSN can be divided into categories[10] such as business (e.g. LinkedIn), common interests (e.g. Learnist, IdeaPlane) dating (e.g. Friendster, Orkut), face-to-face facilitation, friends (e.g. Facebook, Twitter), pets (e.g. Shelter Me, DoggyBNB), and photos (e.g. Instagram, Pinterest).

---

[8] mechanisms that nudge users to consider more carefully the content and context of their disclosures on Facebook.

[9] https://www.techopedia.com/definition/4956/social-networking-site-sns

[10] http://www.socialsoftware.weblogsinc.com/

In [Gross05], Gross and Acquisti demonstrate patterns of personal information revelation, as well as privacy implications that are associated with online networking. These implications depend on the identifiability level of the potential recipients, potential usages and the provided information. A study of information revelation behavior in OSN was conducted, including data related to usage and the inferred privacy preferences of four thousands users on Facebook. It showed that users disclose a huge amount of personal information such as images, date of birth, current relationship status (i.e. single, married), phone number, current residence, political views and various interests.

Privacy in social networks varies depending on people and contexts [VHEA96]. In this direction, research efforts include diary studies [Halderman04, Ito05], surveys [Hawkey06, Spiekermann01] and interviews [House06, Kindberg05]. In [Ahern07], Ahern et al. conducted studies on Privacy Patterns and Considerations in online and mobile photo sharing. Since the amount of camera phone applications is growing fast thus publishing of personal content becomes increasingly easy, users are exposed to new privacy concerns related to their personal and social environment.

Despite its success, social networking services nowadays are increasingly criticized for users' privacy violation [Bonneau09] and personally identifiable information (PII)[11] leakage [Krishnamurthy09]. In [Boyd03, Boyd04] Boyd exhibits issues of trust and intimacy in OSN. Besides these issues, ensuring privacy in social networks constitutes a very critical and important factor. It could be done by setting policies that correspond to the users' expectations. In [Bonneau09], Bonneau et al. argue that providing users with fine controls such that they are able to set their preferences is not sufficient regarding privacy. As a result, they proposed pre-packaged privacy policies that are given to the users in order them to choose whichever fits them. In [Danezis09], Danezis expands this work by proposing a specific privacy mechanism for OSN that is based on context inference through social network analysis. This work introduces policies that are sensitive to the social context in which content is generated and is automatically inferred by the privacy policy mechanism.

[11] information that is used to distinguish an individual's identity, either alone or combined with other public information that is linked to a specific individual

2.3.2 Privacy Nudges for Online Social Networks (OSN)

Privacy nudges are applications of the soft or asymmetric paternalism. Paternalism[12] is "an infringement on the personal freedom and autonomy of a person (or class of persons) with a beneficent or protective intent". In [Calo14] Libertarian paternalism, or "nudging," refers to regulation by exploiting cognitive biases through changes to a physical or digital environment. In [Thaler08], Thaler and Sustein point out that the specific premise of a Nudge is that a familiarity with how people deviate from rational decision-making will help regulators achieving public policy goals without resorting to coercion. They also defined the soft paternalism as "an effort to influence the choices in a way that will make chooser a better, as judged by themselves". Inspired by the concept of the soft paternalism, they define the nudge as a gentle reminder to the user. The formal definition of nudge includes "any aspect of the choice architecture that alters people behavior in a predictable way without forbidding any options or significantly changing their economic incentives".

In the offline world, the user can tailor her comments or hide her actions from a group of people [Goffman59]. However, in online world, once the users share something, then it is visible by everyone. For example, assume a user who is going to share a post under the influence of anger, a nudge cannot prevent her from publishing the content but may warn her about the consequence of her behaviour.

OSN provide users with proper privacy to overcome from context collapse [Wang11]. In the case when data breaches[13] or privacy breaches occur in the OSN, the consequences will be very dangerous and can vary from simple embarrassment to stalking, identity theft or damaged reputations [Boyd07].

[12] http://global.britannica.com/topic/paternalism

[13] https://en.wikipedia.org/wiki/Data_breach

In [Wang11], Wang et al. demonstrate how Facebook disclosures can lead to negative outcomes including personal relationship issues or issues at work place while [Wang13, Wang14] demonstrate Privacy nudges for OSN. In fact, they have developed three privacy nudges for Facebook, due to its popularity and privacy issues' complexity. Their objective is to encourage users to reflect on their post and audience, as well as to prevent them from any online disclosure that they may later regret [Wang11]. These nudges include:

- **Picture Nudge**: This type of nudge has been developed to nudge the user regarding the potential audience of its post. In detail, it selects randomly five people who can see the user post and displays their profile picture whether they are friends or not. Thus, the user knows in advance who can see this post, by seeing these profile pictures. As a result of this nudge, the user can cancel the post and pay attention towards his privacy settings, avoiding the audience with whom user was not willing to share its post.

- **Timer Nudge**: This type of nudge has been developed to provide time to the user before posting. As soon as the user clicks on the button to share something then this nudge adds delay of ten seconds, giving the time to the user to review the content, as well as the option of editing or canceling the post. As the time reaches to three seconds left then another nudge gives the user the option to post the content. However, the user has the option to do the post immediately without any delay, if it finds that it is correct.

- **Sentiment Nudge**: This type of nudge has been developed to provide both immediate feedback on the shared content and delay to the post. Assuming that the user is going to post something on her wall. When the user clicks on the post button, the nudge is going to analyze this content word-by-word, using the AFINN-111 module that includes 2500 words along with their rating. There are three main words' categories: positive, negative and neutral words that are given rating in the range 1 to 5, 0 and -1 to -5, respectively. Thus, the post of the user is analyzed, while the user is nudged whether this post is perceived as positive or

negative. In the latter case, delay is added and the user has the option to edit or cancel the post.

Despite nudges giving several advantages to the users, including actions related to their post, this area has many limitations as well. For instance, timer and sentiment nudge give the option to the user to revise the post content, giving her the sufficient time to correct it and thus avoiding regrettable posts. However, the benefits of the timer nudge come at the cost of the delay. Although, sentiment nudge makes users to be aware of how other users perceive their posts [Weizenbaum66] they do not always work. For example, assuming that a user posts images or video in the OSN rather than text, then the sentiment nudge is not able to nudge the user, since it supports only text recognition of the users' posts. Regarding picture nudge, users are encouraged to be more cautious about their posts, by showing them the profile pictures of the users who are able to see their content. Thus, this nudge warns the users to pay attention to the audience.

Since the idea of nudge is beneficial not only for OSN but also for other domains, some challenging issues related to the privacy nudges [Monteleone15] come up. An example include, a political statement of user considered as positive by the audience but negative by the sentiment nudge. Thus, improvement both related to text understanding and support of multimedia data are needed.

2.3.3 Penumbra: a distributed file system with access control

In [Mazurek14a, Mazurek14b], researchers at Carnegie Mellon University and the University of North Carolina have developed a system that allows the users to apply fine-grained controls to their personal data, while in [Mazurek14a] Mazurek introduces in details Penumbra. Users have content spread across multiple devices and online repositories, such as Web mail, photo sites and social-networking platforms. Penumbra is based on software that runs in the file system and allows the users to set up control on the way they organize and describe their own content by applying tags to this content as well as, by setting policies about who can access files based on those tags.

Penumbra provides users with some predefined categories and rules, giving them the flexibility to define their desired policies based on everyday terms. Moreover, these policies can use automatically generated tags (e.g. based on locations, keywords or facial recognition), since all users are not willing to conscientiously tag all their content. Tags used in Penumbra are cryptographically signed credentials and thus, they can't be forged or spoofed. Each user can assign different tags to the same file. These tags can be based on how they feel about the content while they could also be given reminders about the tags they use most often. The tags themselves can be secret in addition to the files they describe.

Penumbra allows the users to apply access control policies over their files using software that issues challenges among devices to establish a user's identity and authorization to see a requested file. In particular, each user and device is assigned a public-private key pair. The file system intercepts each system call for reading and writing files or metadata, and only allows the system call to succeed if a proof of authorized access can be made. These claims are validated through logical proofs. The proofs are built from authorization credentials that are created by the owner of the file and/or device. Penumbra does not deal with the assigning or authenticating keys. On the contrary, they assume that the keys have already been assigned and the file system matches the user's id with his key. Furthermore, permissions can be cached for a limited time in order to avoid repeating the proof.

Regarding performance of Penumbra, in most cases, users wouldn't notice any lag. In detail, the operations of less than 100 milliseconds are not visible to users, while most of the system calls are under this limit. Assuming the scenario where a user accessing its own files, the system works in about 1 millisecond.

This prototype system was tested on a set of Linux desktops on a LAN, but it could also be implemented across other devices such as tablets and smartphones while it could also be cloud-based storage services.

The Penumbra system along with UCON model concepts inspired us to introduce an algebra (see Chapter 3) that allows users to express Access Control Plans by combining simple operators.

2.4 Personal Data Servers

People would like to have control over their personal information, have freedom; regarding the increasing amount of personal information that is traveling online, as well as the amount of sensitive data that is generated and is used online, concerning their health, finances, energy and time.

As stated in Chapter 1, we implemented a secure data sharing platform for educational purposes. The initial implementation includes secure Token simulation in Raspberry Pi while the final platform involves the real secure Tokens. Moreover, we conducted a sensitive survey experimentation using secure hardware (SMIS secure Token and PlugDB server). In this Section, we review different kinds of personal data servers, due to their important role. Personal servers that have not been used in this work could potentially contribute to future work.

2.4.1Cozy Cloud

Cozy Cloud[14] is a French startup that proposes a "Personal Cloud", revolutionizing the management of personal data. This Personal Cloud is focused on empowerment of the user.

Cozy provides individuals with a personal web deployment platform that can enable them to quickly bootstrap applications and interact with their data. It runs on a server - between the applications and the operating system – simplifying system administration, web development and security.

[14]https://cozy.io/fr/

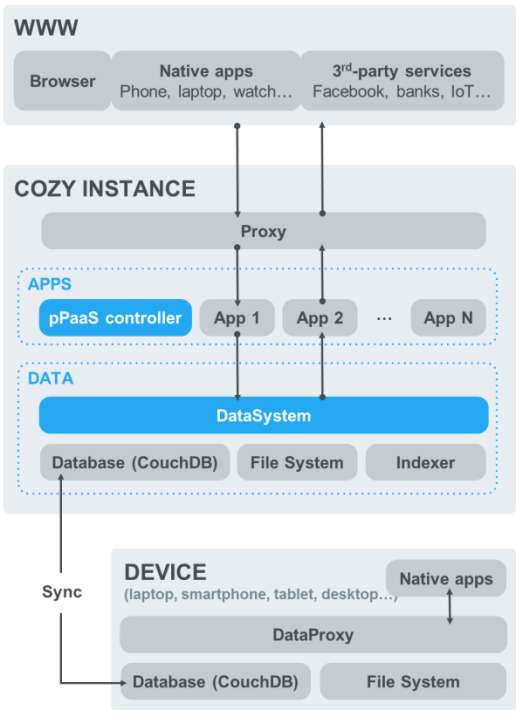

Figure II -9. Cozy Architecture

Cozy proposes interoperable by-design architecture that consists of three main components:

- **Proxy**: It handles user authentication and request redirections to the applications.
- **Data System**: It is the heart of Cozy. It is the data persistence service, allowing data sharing between the applications (i.e. contacts, calendar, banking data, bills data, and pictures) as well as application reaction when data is changing. Cozy architecture is "user centric" and thus also "data centric". CouchDB[15,16] is the database that is used. It is a no-SQL document-oriented database that can be

[15] https://fr.wikipedia.org/wiki/CouchDB

[16] http://couchdb.apache.org/

synchronized with databases that are installed on users' terminals, enabling the "continuity of the digital context" on the users' terminals.

- **Personal Platform as a Service (PaaS)**: It provides an execution environment and enables applications' collaboration involving personal data. It consists of the controller and a command line interface. Controller installs, runs, updates and removes applications within Cozy.

Thus, Cozy is a personal server (Personal Cloud), running on any kind of equipment, either a computer terminal, Internet Box or any other machine. Once the connection to Cozy is established, icons and widgets for services that have already been installed are available. Cozy is Open Source, so that third parties can contribute by developing applications and adding them in the Cozy's market place. Such applications therefore are available to be installed.

Cozy can be characterized as individuals' digital world, facilitating the way that they use and protect their data. Some examples include:

- **Single Sign On**: a single connection is needed in order for the user to access her data (i.e. specific folder, agenda's information).

- **Global Search:** a single user's request can access all her data.

- **Applications' Integration:** The applications are independent of each other. However, all applications use the same data, hence for the same data any update on applicationX is reflected on applicationY.

- **Unified Data:** the user uses one single contacts' database, regardless of the chosen account. Indeed, users can import their contacts from Gmail or LinkedIn in order to benefit from Cozy services.

2.4.2 own Cloud

ownCloud[17] is open source software that provides individuals, enterprises as well as service providers with file synchronization and sharing services, proposing an ownCloud server that protects and manages files within the ownCloud environment, including file storage, user provisioning and data processing.

ownCloud monitors all data access events for downstream auditing and analysis while the server provides the administrators with a secure web interface for controlling ownCloud‘s resources (i.e. backup), managing users’ authorization to set policies and enable or disable features (i.e. firewalls). In ownCloud, an encryption module is available to provide an additional layer of encryption for user files. Features of ownCloud include an online text editor, virus scanner, file versioning and server-side encryption. More advanced features can include enhanced logging, audit plug-ins, File Firewall, SAML authentication.

At its core, ownCloud is a PHP web application that can manage any aspect of it, including user management, plug-ins, file sharing and storage. It is running on top of IIS or Apache on either Windows or Linux. There is a database attached to this PHP application where ownCloud stores users, user-shared file details, plug-in application states, and the ownCloud file cache (a performance accelerator). Moreover, there is an abstraction layer, allowing ownCloud to access the database and enabling support for Oracle, MySQL, SQL Server, and PostgreSQL. Webserver logging is supported via webserver logs. Dynamically allocated storage driven by user directory entries is included in user configurations, enabling data segregation and multi-tenant deployments.

Existing APIs of ownCloud that can be integrated to other systems include:

- **Activity**: This API provides a RSS feed or delivers activities associated with a user‘s file (i.e. sharing, updating, renaming, deleting)

[17] https://www.owncloud.com

- **Applications**: This API provides extensions of ownCloud through integration with existing infrastructures and systems, and the creation of new plug-in applications (i.e. authentication back ends, music and video streaming applications)
- **Capability**: This API includes information related to installed ownCloud capabilities.
- **External provisioning**: This API supports addition and deletion of users remotely, as well as storage metering by administrators.
- **Sharing**: This API allows external apps (i.e. mobile app) to share files from remote devices.

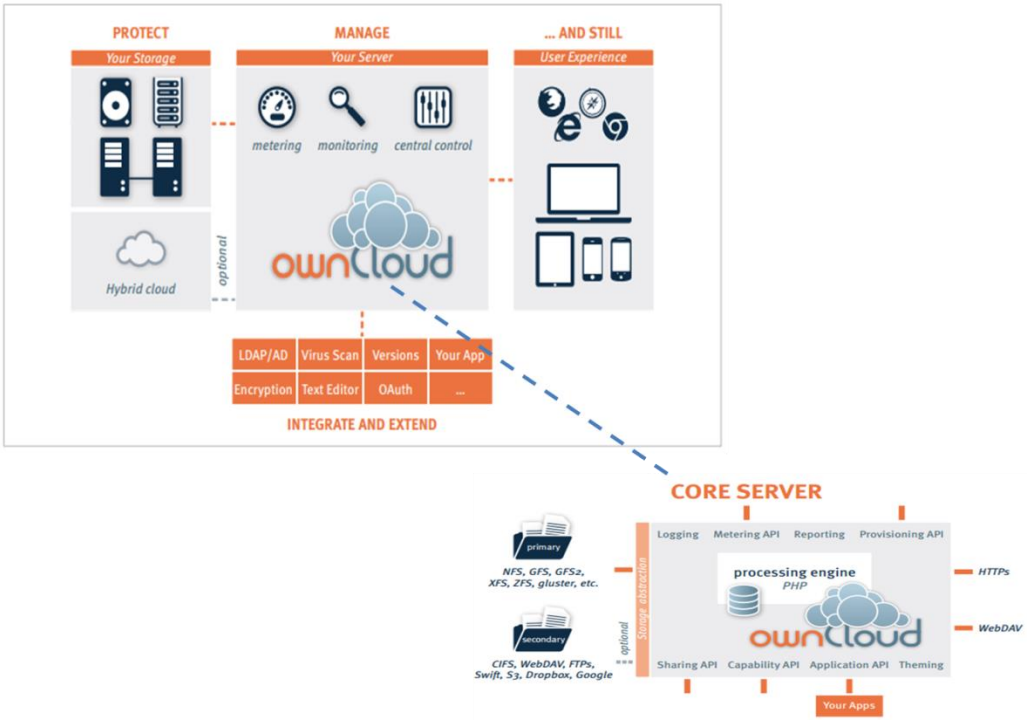

Figure II -10. ownCloud Architecture

2.4.3 Raspberry PI

The Raspberry Pi[18](see Figure II - 11) is a tiny computer that is designed as a platform for educational purposes, for expansion and technological enlightenment, as well.

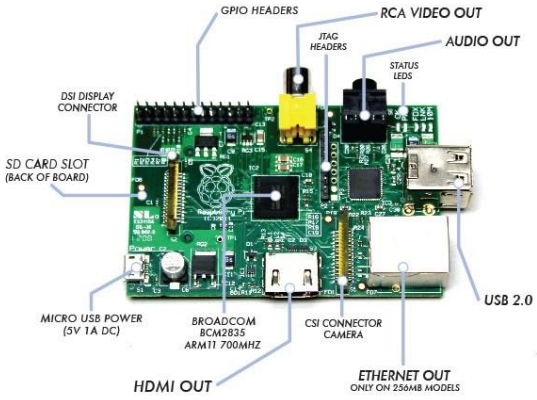

Figure II - 11. Raspberry PI Architecture

There are two different models of Raspberry PI[19,20] (see Figure II - 12), the Model A (for $25) and the Model B ($35), respectively. Model B have introduced three different types of the same model (see Figure II – 12b, Figure II – 12c, Figure II – 12d). The main differences between them are described below.

Model A and Model B Pi 1 are Broadcom BCM2835 systems on a chip, including ARMv6 single core CPU 700MHz, with a Dual Core VideoCore IV GPU and 256MB and 512MB of SDRAM, respectively. Model B Pi 2 and Pi 3 are Broadcom BCM2836, including ARMv7 quad core CPU 900MHz and 1.2GHz, respectively with a Dual Core VideoCore IV GPU and 1GB of SDRAM.

[18]http://en.wikipedia.org/wiki/Raspberry_Pi#Specifications
[19]http://www.trustedreviews.com/opinions/raspberry-pi-3-vs-pi-2
[20]http://www.makershed.com/pages/raspberry-pi-comparison-chart

Figure II - 12.Raspberry Pi Models (Model A and Model B)

Model A is equipped with one USB 2.0 port while Model B(see Figure II - 12) with two (Raspberry Pi 1) or four (Raspberry Pi 2 and Raspberry Pi 3), according to their type, for getting external data. On the wireless side, Raspberry Pi 3 Model B supports Wi-Fi and Bluetooth connections. Additionally, there is a 3.5mm audio output, two video outputs, a standard RCA video port for achieving display connection, while an SD/MMC/SDIO card slot is also available.

GPIO interface is available in order to connect the device to various electronics gadgets and appliances.

In order to use the Raspberry Pi[21], an installation of bootable operating system i.e. Raspbian (Debian GNU/Linux for Raspberry Pi) and NOOBS software[22] (Java SE Platform Products, licensed under the Oracle Binary Code License Agreement) onto an SD card of 4GB or greater is required[23].

[21] http://www.raspberrypi.org/
[22] downloads.raspberrypi.org/noobs
[23] https://www.sdcard.org/downloads/formatter_4/eula_windows/

Some of the applications using the different model types of Raspberry Pi include:

1. **Servers:** This general term include the several types of servers that we will describe in details later. In the context of a server, the Raspberry Pi can be connected and running continuously without any interruption. It will probably remain close to the internet box and will be controlled remotely via SSH. However, servers require a strong ability to manage scalability (e.g. significant amount of resources, many users connect simultaneously). For example, this means that these servers require SD cards that allow fast writing, and a Raspberry Pi Type B, with more RAM. However, they do not normally require Wi-Fi, or wireless keyboard.

   In the context of servers, we can distinguish several broad categories:

a. **The web server:** This is a type of a server to host one or more web sites, which are accessible from everyone. These servers are often used by developers that are willing to manage the whole configuration of their server, without investing a lot in a professional web host for a dedicated server, as well as without buying a real very expensive, noisy server. Moreover, in a UNIX system or GNU / Linux, Raspberry Pi allows the installation of a web server easily. For this use, Raspberry Pi can be a powerful tool for Web developers, or students.

b. **The storage server:** This is a type of a server (known as FTP server[24)] that stores any files accessible to the owners from anywhere. This type of application has relatively larger audience than the web server. However, in some cases people prefer the use of an SFTP connection, rather than FTP where the file transferring is achieved via SSH Protocol. Raspberry Pi, type A and B can contribute to build this type of server.

---

[24] a machine running a software to handle the protocol "File Transfer Protocol": a protocol dedicated to the file transfering.

c. **The service server:** This is type of server is designed to accommodate a service (e.g. to access it from anywhere). The term "service" signifies software, running at the user's computer, making calls to a central entity. This category may include game servers, but also those providing public services, such as VPN servers or time servers (e.g. for machines' synchronization). Raspberry Pi, especially Type B can act as this kind of server.

2. **The multi-media use:** The idea of advanced multi-media, mostly done in the media-center is one of the most developed applications of Raspberry Pi. A media-center is a computer system, being grafted on television, providing supplementary TV services (to the basic ones) such as a more user friendly interface, internet connection, the ability to store and to play music, pictures, videos, and possibly other types of files. However, the media-centers are relatively expensive, and thus the Raspberry Pi media-center application could be a cheaper solution to this direction.

   For this kind of application, XBMC software is used. XBMC software allows playback of various video formats, playlist creation, internet connection and many other features, including the ability to add Python scripts to XBMC in order to be enhanced.

3. **Home automation:** Home automation includes the techniques that can centralize the management of various facilities of a house. These facilities include heating management, data collection (e.g. to better manage energy costs), turn on or off lamps remotely. Thus, individuals can be able to control their home via a computer.

   However, existing frameworks to support home automation are quite expensive. Raspberry Pi Type B can accommodate numerous modules to communicate with home automation equipment. There are also multiple interfaces for Raspberry dedicated to home automation, including Domoticz[25].

[25] https://domoticz.com/

4. **Embedded systems:** Raspberry Pis are often used in the electronics-related projects, as central controllers of a computer system. Many projects are emerging, such as controlling a remote-controlled car with cameras.

5. **A computer:** Raspberry Pi is a computer, small, cheap, allowing programming as well as, encouraging learning. Some examples include programmers that they need a Linux machine for a particular reason (without particular computing power demands) such as navigation, checking emails, and occasionally watching videos.

To sum up, indeed, Raspberry Pi is often used as a server, as a media-center, as home automation manager but it is primarily a computer, with many functionalities, being cheaper compared to other computers.

In view of the above considerations, Raspberry Pi provides people and academic communities with a cheap platform for small-scale systems that is ideal for educational purposes. However, the SD interface is a bit slow as well as 100Mbit Ethernet gets rarely full speed. An SSD hard drive or USB drive can contribute to a better performance.

We simulate the environment of Secure Portable Tokens (see Section 4.5) in Raspberry Pi 2 model B, in order to build an initial prototype data sharing platform (see Chapter 4 - Section 1).

2.4.4 Freedom Box

FreedomBox is a consumer electronics device (secure platform) that is easy to setup, maintain and use. Some of the services that this platform provides include file sharing, shared calendar, instant messaging, secure voice conference calling, blog and wiki, focusing on confidentiality. FreedomBox supports being installed in a variety of power-efficient hardware (i.e. single board computers or other devices), while its installation could also be available to any virtual machine, by installing the FreedomBox-setup package.

Figure II - 13.FreedomBox Supported Hardware

Figure II-13 shows the supported devices including the Olimex A20 OLinuXinoLime 2[26] (see Figure II - 13.d), the BeagleBone Black[27] (see Figure II - 13.c), FreedomBox Danube (Figure II - 13.a), Cubietruck (see Figure II - 13.b).Closed-source boards like the DreamPlug[28] (see Figure II - 14.a) and the Raspberry Pi[29,30] (see Figure II - 14.b,Figure II - 14.c,Figure II - 14.d) are also possible options, but not recommended due to performance/cost concerns.

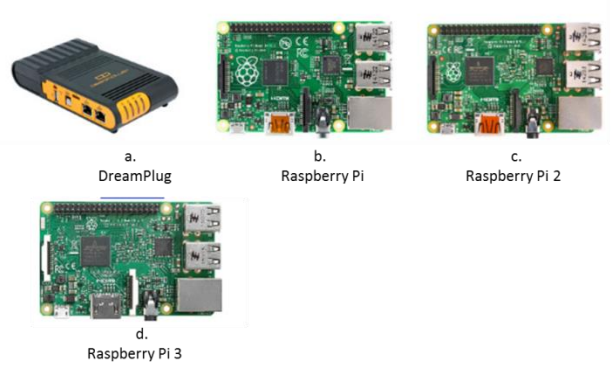

Figure II - 14.FreedomBox Alternative Supported Hardware

[26]"FreedomBox/Hardware/A20-OLinuXino-Lime2 - Debian Wiki".wiki.debian.org. Retrieved 2015-11-22.
[27]"FreedomBox/Hardware/BeagleBone - Debian Wiki".debian.org.
[28]"FreedomBox/TargetedHardware - Debian Wiki".debian.org.
[29]"FreedomBox/Hardware/RaspberryPi - Debian Wiki". debian.org
[30]FreedomBox/Hardware/RaspberryPi2 - Debian Wiki". debian.org

Once the desired hardware is chosen and the FreedomBox installed, setting up and operating a FreedomBox is quite easy (like setting up a smart phone).

Since FreedomBox promotes decentralized deployment of hardware, it is quite promising to provide privacy in the everyday life, as well as secure communications especially for the individuals who seek to preserve their freedom in oppressive regimes.

2.4.5 Personal Data Servers: Secure Portable Token

SMIS (Secured and Mobile Information Systems) research is focused mainly on the privacy's protection of personal data. In [Allard10], T. Allard et al. have described a vision of Personal Data Server (PDS) which aims to build a credible alternative to the systematic centralization of personal data on servers. The main idea behind PDS is to embed on secure hardware, such as smart cards with great storage capacity, software components that are able to acquire, store and manage various types of personal data such as images, invoices, bank statements, medical data and location traces. These software components are intended to constitute an effective personal data server that is able to interoperate with other external servers or services giving to the data holder total control over his personal data.

PDS safety is derived from the secure hardware in which it is embedded. Since, there are a wide variety of secure components (e.g. mass storage SIM card, secure USB key, secure portable token, smart dongle), all can be abstracted by: a Trusted Execution Environment and a large storage space that is potentially untrusted and therefore contains only encrypted data. The Trusted Execution Environment is usually comprised of a tamper-resistant secure microcontroller that is resistant to physical attacks, while the storage space usually consists of an external Flash memory (e.g. Flash chip or card micro-SD). The PDS engine is executed in the Trusted Execution Environment while the individual's data is hosted cryptographically protected in the external Flash. This engine is an effective Database Management System (DBMS) that is embedded in a secure hardware device. It is able to store data as tables, and apply operations over these tables such as: indexing or querying via SQL queries. It also guarantees the logical (integrity constraints) and physical (transactional atomicity) integrity of the tables and

provides protection via access control. The implementation of such embedded DBMS raise many scientific obstacles related to hardware constraints caused by the microcontroller (e.g. tiny RAM) and NAND Flash (e.g. cost of random rewrites, Block-erase-before-page rewrite). Thus, a deep redefinition of the classical principles of database management (e.g. storage, indexing, requests, transactions) [Allard10] is required, in order to overcome these obstacles.

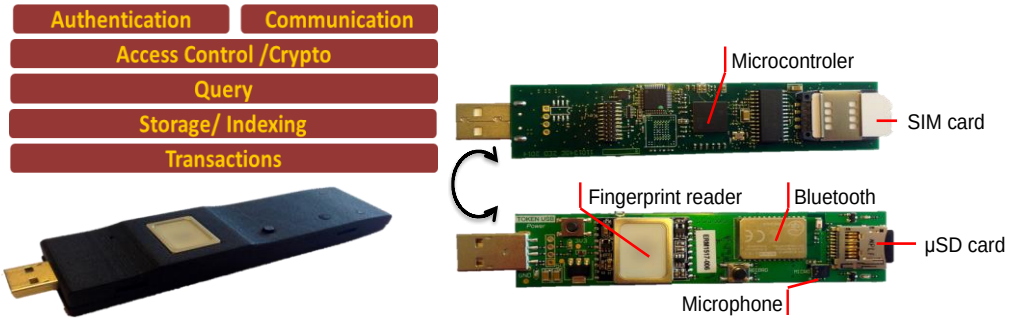

Figure II - 15.Software and hardware architecture of Personal Data Server

PDS are equipped with tamper-resistant microcontroller that makes physical attacks difficult. It discloses data of only a single individual. These facts reduce significantly the cost / benefit ratio of an attack and thus minimizes the interest of such an attack. It is pluggable on demand, self-administered, preventing physical attacks while it constitutes a secure repository where the stored data can be accessed upon owner's authentication and following user's access control rules. Considering distributed treatments between users, this ensures that an individual can export data from its PDS to a partner's PDS without risk of loss of confidentiality, involving usage control rules such as Sticky Policies [Mont03, Tang08, Chadwick10, Kelbert12] that will be handled by the PDS of that partner.

In [Anciaux15a, Anciaux15b] an operational PDS prototype (PlugDB) has been developed. This prototype is embedded in a secure portable token (SPT) that can

combine hardware security and large quantities of NAND Flash memory storage in a portable form factor. It is a full-fledged data server (PlugDB server) (see Figure II - 15), running on any device that is equipped with USB port or Bluetooth, such as personal computers, tablets and smartphones. Such devices, with adequate software, allow their owners to manage and control their sensitive data. It provides security guarantees and can be used without network connection. PlugDB is used for experimentation, involving portable and secure medical and social records, facilitating the coordination of care at home for dependent persons in the territory of Yvelines [Anciaux08]. PDS vision has evolved toward a broader vision called Trusted Cells, including personal data, coming from the Internet of Things [Anciaux13].

It is obvious that the role of such secure devices like SPT is decisive since they can serve as an answer to privacy deficiencies in different sectors of life (e.g. education, transportation or healthcare systems) and contributes variously towards this thesis. Firstly, SPT have been used in order to test our data sharing platform, as it is described in Chapter 4. Consequently, in the same Chapter we discuss about an experimentation that has been conducted, involving one hundred forty non-expert users answering a questionnaire survey in the context of job-seeking, using this secure device.

It is worth mentioning that the different notations that will be used throughout this thesis regarding either secure hardware or software, include the following terms: Secure Personal Token (SPT), Token to describe a secure device on which a Personal Data Server is embedded and running on it while TrustedCell, PlugDB server, PDS refer to the corresponding software that is running in the devices giving individuals the control over their sensitive data.

2.5 Conclusion

In this Chapter, we first overviewed some well-known Access Control Models. Then, usage control was introduced. After listing several types of usage control models, we emphasized on $UCON_{ABC}$ model and its enhancements. Moreover, we reviewed the domain of OSN in terms of existing sharing policies as well as the mechanisms that have currently been developed to enhance and strengthen privacy on OSN. We also

presented an existing system, named Penumbra for fine-grained controls over personal data. Finally, we surveyed the state of the art related to personal data servers both on hardware and software side.

Chapter III
Data Sharing of Personal Data

In this chapter, we start by a quick comparison of basic access and usage control models. Having noticed the limitations of access control models and the efficiency of $UCON_{ABC}$ model, we introduce in this chapter two implementations of this model. The former implementation involves a usage control engine that could act as a service within a system (e.g. reference monitor). In order to simplify the former implementation, the latter implementation includes a data sharing application that is benefited from $UCON_{ABC}$ engine using an SQL Database. A graphical user interface (GUI) has been designed as well. However, this implementation is restricted to data sharing between two users. Nowadays, the amount of the individuals that use the OSNs is rising; the amount and the diversity of personal data that is exchanged are increasing as well. In addition, the variety of users with whom individuals share this data is increasing. As a result, there is a need of extending these implementations in order to cover the cases of OSN. We thus introduced DatShA, a data sharing algebra that allows individuals to choose how they exchange their data, since it provides an infrastructure along with a set of generic operators. Thus, individuals using this infrastructure can define Access Control Plans (ACP) by combining these operators and thus can manage the accessibility and usage of their data. This work was published in conference proceedings of EDBT/ICDT 2016 Joint Conference [Bouganim16].

3.1 UCON model Implementation

In Chapter 2, we discussed in details some Access and Usage Control models. Table III – 1 presents a comparison of the models described according to some important characteristics, including:

- **Expressiveness:** The model allows individuals to specify access rights for a diversity of objects and granularity levels based on specific features such as roles, context, and location.
- **Concurrency:** The model allows individuals to access and share their data between them, concurrently.

- **Continuity of Control**: The model supports enforcement of access control before, during and after access is granted.
- **Mutability:** The model allows users to specify and modify policies depending on the environment and mutability of attributes.
- **High-level Access Rights:** The model allows users to specify access rights in high level; hence, users are able to manage complex fine-grained policies.

According to Table III - 1, it is evident that $UCON_{ABC}$ is a powerful usage control model since it covers the majority of the features described. As a result, two implementation of this model have been developed. The former implementation has been done in Java using PolicySet, Rules and Schema of eXtensible Access Control Markup Language (XACML) while the latter implementation takes advantage of a relational Database in order to extend and simplify the former one.

Access & Usage Control Models	Model's Characteristics				
	Expressiveness	Concurrency	Continuity	Mutability	High level Access Rights
MAC, DAC, RBAC	Partially	No	No	No	Yes
TM, DRM	Yes	No	No	No	Yes
$UCON_{ABC}$	Yes	Partially	Yes	Yes	Yes

Table III - 1. Comparison of Access and Usage Control Models

3.1.1 $UCON_{ABC}$ model: Design using XACML

This section presents the $UCON_{ABC}$ engine which was developed as a general engine that could be used into an application (e.g., social network application) or as a service within a system (e.g., reference monitor), to enforce usage control policies on objects. This implementation is based on the description of the model [Park02, Park04] by Jaehong Park, using the Schema of eXtensible Access Control Markup Language

(XACML)[1,2,3] and XML[4,5,6,7,8] [Piez99] in order to express access control rules and conditions.

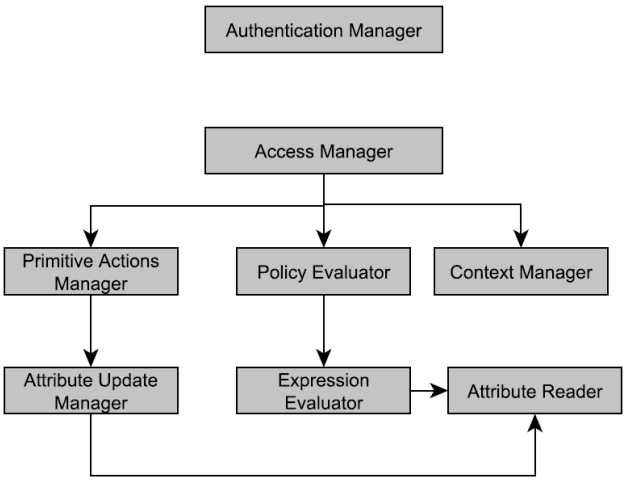

Figure III - 1. Modules of UCON engine

We have distinguished the different parts of the UCON model (see Figure III - 1), in order to implement the UCON engine. Authentication Manager authenticates the subject, based on subject's ID and subject's passphrase while Access Manager is designed to manage and regulate access to objects by processing subjects' requests to exercise rights on objects. Moreover, it decides if the access is going to be granted or denied. Policy evaluator allows this decision to be taken, while Context Manager is focused on keeping track of the active usage sessions on an object. The feature of $UCON_{ABC}$ model known as Continuity of Access Decision, as we described in Chapter 2, is implemented by the Context Manager module (see Figure III -1) while the updates of

[1] http://sunxacml.sourceforge.net/
[2] http://gryb.info/xacml/doc/xacmlightreference.html
[3] http://xml.coverpages.org/xacml.html
[4] en.wikipedia.org/wiki/java-api-for-xml-processing
[5] https://www.oasis-open.org/committees/download.php/2713/
[6] stackover_ow.com/questions/373833/best-xml-parser-for-java
[7] www.mkyong.com/java/how-to-read-xml-_le-in-java-dom-parser
[8] xerces.apache.org/xerces-j/.

the attributes is going to be done by Primitive Actions Manager. Attribute Update Manager manages Mutability of Attributes of this model.

The attributes are stored and updated in XML files. These files contain subject and object attributes. Attribute Reader retrieves subjects' and objects' attributes. The values of these attributes are very important for the expressions' evaluation during the policies' evaluation and the attribute update process, as well. Policy Evaluator retrieves and parses policies that are associated with the usage of an object. Expression Evaluator evaluates the expressions that are fundamental for the evaluation of usage policies.

The entries that are related to the usage sessions along with the attribute updates that are taking place within a usage session are stored in a log file (see Figure III - 2). Thus, this file keeps the information for the usage of the object and enforcement of the UCON policy (XML structure), as well.

An event includes information for the start and the end of usage, as well as the state transition and the attribute updates. This information is recorded in the log file along with the time stamp, the subject's id, the object's id and the right that the subject has requested to exercise on the object. Moreover, information that is related to the state transition is stored in the log file. In this case, we keep either the time stamp or the previous state. We also keep the type of the attribute update (preUpdate, postUpdate, or onUpdate), the object's ID, the new value of the object attribute and the previous value of the object attribute to verify the attribute updates.

```
<?xml version="1.0" encoding="UTF-8"?>
<log:UsageLog xmlns:log="ucon/UsageLog"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="ucon/UsageLog/UsageLog.xsd ">
  <log:UsageSessions>
    <log:NewUsageSession Object="Document" Right="Read" Subject="alice001" TimeStamp="2014/04/04 16:48:09"/>
    <log:AttributeUpdate AttributeName="NumberOfTimes" OldValue="[0.0]" TargetObjectID="Document"
                         TimeStamp="2014/04/04 16:48:09" UpdateType="PreUpdate" UpdatedValue="[1.0]">
        <log:ExpressionEvaluated FunctionID="katsurak.ucon.functions.Add" Result="[1.0]" TimeStamp="2014/04/04 16:48:09">
            <log:Inputs>
                <log:Attribute AttributeName="NumberOfTimes" ObjectID="Document" Value="[0.0]"/>
                <log:Constant>1</log:Constant>
            </log:Inputs>
        </log:ExpressionEvaluated>
    </log:AttributeUpdate>
    <log:StateTransition CurrentState="Requesting" PolicyEvaluated="Requesting" PreviousState="NotApplicable"
                         SessionID="alice001,Document,Read" TimeStamp="2014/04/04 16:48:09"/>
    <log:StateTransition CurrentState="Accessing" PolicyEvaluated="PermitAccess" PreviousState="Requesting"
                         SessionID="alice001,Document,Read" TimeStamp="2014/04/04 16:48:10"/>
    <log:EndUsageSession Object="Document" Right="Read" Subject="alice001" TimeStamp="2014/04/04 16:49:28"/>
  </log:UsageSessions>
</log:UsageLog>
```

Figure III - 2. Part of usage log file

The policies in the UCON model are referred to a set of policies for each state. For instance, the following terms: requestingAccess, grantedAccess, deniedAccess, revokedAccess and endAccess, could describe a state, according to the model's description in Chapter 2. Figure III - 3 illustrates the different policies that can be executed. Each policy includes a target that identifies the subject, the object and the right for which the policy is going to be enforced.

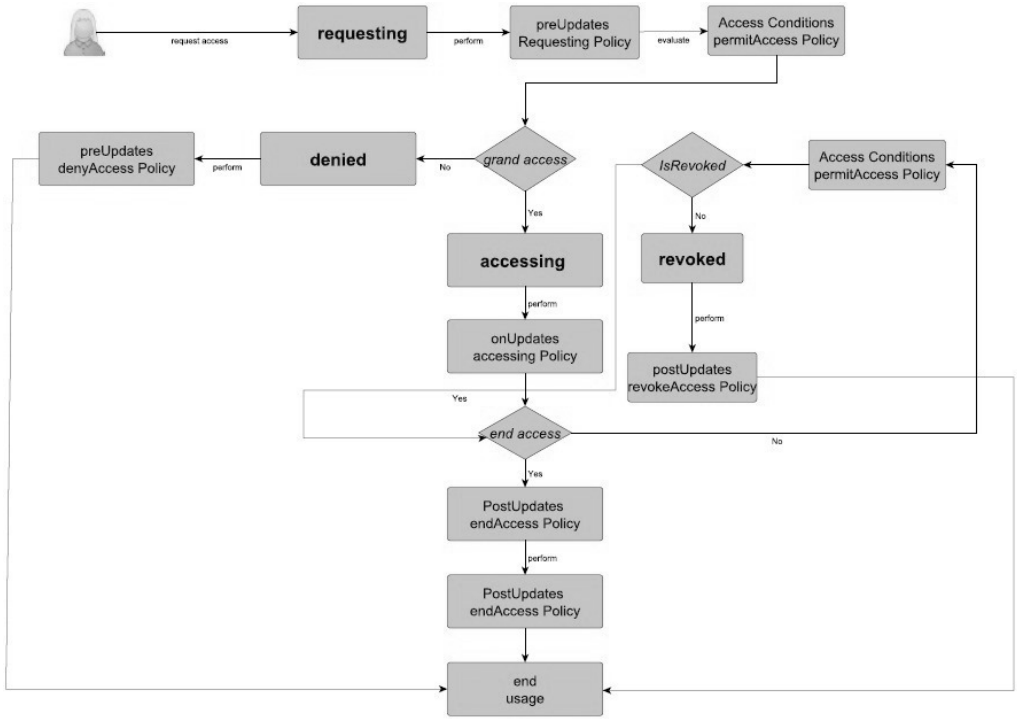

Figure III - 3.Control flow and execution of different policies

Requesting Policy includes the PreUpdates that will be performed when the request is made. The transition from requesting state to accessing state is controlled by a policy. This policy includes the conditions under which a subject can access an object for a specific right. When the conditions in this policy are met in the requesting state, the subject's state is the accessing state and attribute updates associated with this state are performed. These updates are called OnUpdates and are continuously performed while the subject is accessing the object.

When the conditions are not met in requesting state, the subject cannot access the object and thus the access denied state occurs. The attribute updates associated with this state are performed. The conditions for accessing the object are checked continuously while the subject is accessing an object. When the conditions can no longer be met due to some modifications, then the access is revoked. The attribute updates that are associated with the revoked state are performed. The subject could stop accessing the object before access is revoked and the endAccess state happens and the associated attribute updates are performed.

As we discussed before, there are four states in which a subject could be transited; including requesting state, accessing state, denied state and revoked state. There are various policies that can be defined for the usage of an object. Those policies are written in XML, according to the Schema of XACML.

More precisely, Requesting Access Policy consists of the attribute updates that are to be performed when the requesting state happens. Permit Access Policy consists of the conditions that should be met before accessing state. The evaluation of this policy is taking place during the accessing state. When the conditions are no longer met, then the access is revoked. Accessing Policy includes the updates of attributes that are performed during the accessing state. Access Denied Policy includes the updates of the attributes that are performed in the access denied state. Revoke Access Policy includes the updates of attributes that are performed in the access revoked state. Finally, End Access Policy contains the attribute updates that are performed after the user ends the access.

A schema for the UCON policy has been developed. The policy file includes the target of Policy, the Type of Policy (see Figure III - 4), the Conditions, the Primitive Actions, the Attribute Updates and some Expressions. The target of the policy, as we mentioned above, identifies the subjects, objects and the rights for which the policy is applicable. The policy type determines the state of the usage session at which the policy is evaluated.

The condition element is used to determine the conditions under which the state transition into accessing state is allowed and can be determined by expressions. The primitive actions include attribute updates for implementing the concept of mutability of attributes, as we discussed in Chapter 2. Finally, some expressions are used in the definition of conditions and attributes' updates in a policy.

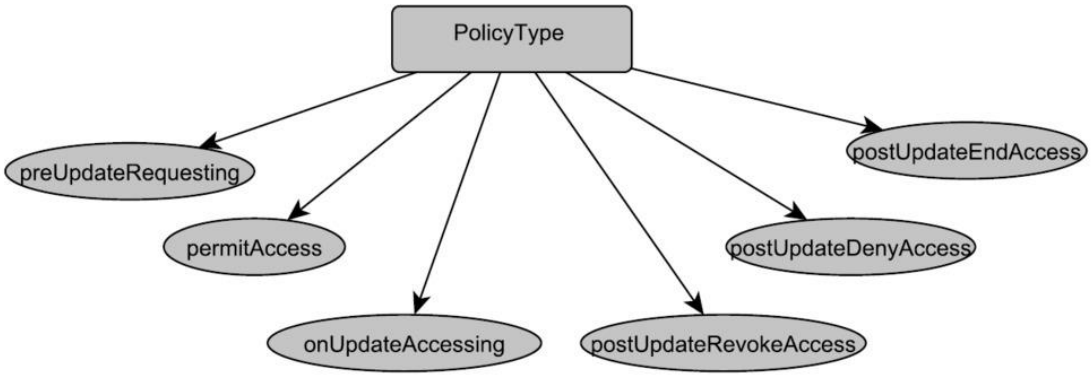

Figure III - 4. Type of Policy

3.1.1.1 Scenarios

In this section, we are going to present an example in order to show the power of the UCON engine and show how UCON models can be applied to protect the privacy of digital information.

Suppose a user Alice (subject) (see Figure III - 5) would like to exercise the right "Read" on a Document (object) (see Figure III - 6). This user works in the Computer Science Department (attribute) as a developer (attribute) and she would like to access the Document twice (attribute) during the office hours (attribute).

Thus, the user has a unique subjectID (alice001) along with a subjectPassphrase (mySecretPassPhrase) for authentication purposes. We defined the subject's attributes as follow: "Department" with value "ComputerScience, "DesignerOf" with value "UCON model", "Name" with value "Alice" and "DocumentReadTime" with value "12:00". Similarly, the Document (object) that the user is going to access has a unique objectID (Document) (see Figure III - 6). We defined the object's attributes as follows: "TimeForRead" with value "12.00" and "NumberOfTimes" with value "2.0".

The AuthenticationManager is going to authenticate the subject while the ContextManager maintains a Map of subjectID (key) which allows storing the subjects' and objects' attributes that will be accessible to the subject along with the corresponding rights (in this case: "Read"). The AccessManager takes the decisions for allowing or

denying the access to the subject (i.e. through events). Reading and Updating of attributes are performed by the AttributeManager.

```
<?xml version="1.0" encoding="UTF-8"?>
<subj:Subject xmlns:subj="ucon/Subject"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="ucon/Subject/Subject.xsd ">
    <subj:SubjectID>alice001</subj:SubjectID>
    <subj:Attributes>
        <subj:Attribute>
            <subj:AttributeName>Department</subj:AttributeName>
            <subj:AttributeValues>
                <subj:AttributeValue>ComputerScience</subj:AttributeValue>
            </subj:AttributeValues>
        </subj:Attribute>
        <subj:Attribute>
            <subj:AttributeName>DesignerOf</subj:AttributeName>
            <subj:AttributeValues>
                <subj:AttributeValue>UCONmodel</subj:AttributeValue>
            </subj:AttributeValues>
        </subj:Attribute>
        <subj:Attribute>
            <subj:AttributeName>Name</subj:AttributeName>
            <subj:AttributeValues>Alice</subj:AttributeValues>
        </subj:Attribute>
        <subj:Attribute>
            <subj:AttributeName>DocumentReadTime</subj:AttributeName>
            <subj:AttributeValues>
                <subj:AttributeValue>12:00 AM </subj:AttributeValue>
            </subj:AttributeValues>
        </subj:Attribute>
    </subj:Attributes>
    <subj:Secret>MySecretPassPhrase</subj:Secret>
</subj:Subject>
```

Figure III - 5. XML file with subject's information

```
<?xml version="1.0" encoding="UTF-8"?>
<obj:Object xmlns:obj="ucon/Object"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xsi:schemaLocation="ucon/Object/Object.xsd ">
    <obj:ObjectID>Document</obj:ObjectID>
    <obj:Attributes>
        <obj:Attribute>
            <obj:AttributeName>TimeForRead</obj:AttributeName>
            <obj:AttributeValues>
                <obj:AttributeValue>1200</obj:AttributeValue>
            </obj:AttributeValues>
        </obj:Attribute>

        <obj:Attribute>
            <obj:AttributeName>NumberOfTimes</obj:AttributeName>
            <obj:AttributeValues>
                <obj:AttributeValue>2</obj:AttributeValue>
            </obj:AttributeValues>
        </obj:Attribute>
    </obj:Attributes>
</obj:Object>
```

Figure III - 6. XML file with object's information

Figure III - 7 shows the policy file where the user Alice (subject) can read (right) the Document (object), if she works at the Computer Science Department, from 12:00 until 13:00. The Expression Evaluator evaluates the expressions while the Policy Evaluator evaluates the policies and the attributes updates. The Access Manager creates the decisionPolicyEvent and the mutabilityEvent for every object's request. The conditions are checked continuously for any modifications.

```
<?xml version="1.0" ?>
<p:PolicySet xmlns:p="ucon/UCONPolicy"
             xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
             xsi:schemaLocation="ucon/UCONPolicy/UCONPolicy.xsd ">
    <p:Policy PolicyType="PermitAccess">
        <p:Target>
            <p:Subject> alice001 </p:Subject>
            <p:Object>  Document </p:Object>
            <p:Right>   Read     </p:Right>
        </p:Target>
        <p:Condition CombiningAlgorithm="katsurak.ucon.functions.All">
            <p:Expression FunctionID="katsurak.ucon.functions.LogicalAnd">
                <p:Expression FunctionID="katsurak.ucon.functions.StringEqualIgnoreCase">
                    <p:ObjectAttribute>
                        <p:RequestingSubject/>
                        <p:AttributeName>Department</p:AttributeName>
                    </p:ObjectAttribute>
                        <p:Constant DataType="String">ComputerScience</p:Constant>
                </p:Expression>
                <p:Expression FunctionID="katsurak.ucon.functions.LessThan">
                    <p:ObjectAttribute>
                        <p:RequestedObject/>
                        <p:AttributeName>NumberOFTimes</p:AttributeName>
                    </p:ObjectAttribute>
                        <p:Constant DataType="Integer">2</p:Constant>
                </p:Expression>
                <p:Expression FunctionID="katsurak.ucon.functions.LogicalAnd">
                    <p:Expression FunctionID="katsurak.ucon.functions.TimeGreaterThan">
                        <p:Expression FunctionID="katsurak.ucon.functions.EnvironmentVariable">
                            <p:Constant DataType="String">CurrentTime</p:Constant>
                        </p:Expression>
                        <p:Constant DataType="Time">1200</p:Constant>
                    </p:Expression>
                <p:Expression FunctionID="katsurak.ucon.functions.TimeLessThan">
                    <p:Expression FunctionID="katsurak.ucon.functions.EnvironmentVariable" >
                        <p:Constant DataType="String">CurrentTime</p:Constant>
                    </p:Expression>
                        <p:Constant DataType="Time">1300</p:Constant>
                    </p:Expression>
                </p:Expression>
                </p:Expression>
            </p:Condition>
    </p:Policy>
</p:PolicySet>
```

Figure III - 7. UCON Policy

3.1.2 $UCON_{ABC}$ model: Design using SQL Schema

In this section, we propose an SQL schema, as an approach to simplify the previous implementation of the section 1.1.

This implementation includes an application for Data Sharing (i.e. load/edit images) in which we used the new UCON Engine that we developed using SQL[9] [DBS, Matthews03, SQLManual, SQLiteManual, SQLW3C] and Java. This application could be a part of a Social Network, in which the user can view or/and edit the images of other users (right: load or edit), according to the permissions of the owner of the image. In our case, an SQL Schema and some functions which represent the UCON Engine have been developed. These functions check the conditions that have to be fulfilled to decide if the access is going to be granted or denied. A location Database keeps pairs of IPs and country prefixes, in order to include contextual information to our conditions. A graphical user interface (GUI) has been designed as well. Our goal has been to develop an application, in which the UCON Engine is embedded in order to show the power of this model, through some scenarios.

Our proposed SQL Schema consists of tables which are related to the subjects and the objects as well as their associated attributes, the rights that subjects can exercise on the different objects, the policy rules that should be satisfied, mutation features and continuity of decisions features.

In the rest of this section, we present the SQL schema, the UCON database, the Location database and the functions that regulate the access, in detail.

3.1.2.1 SQL Schema, UCON and Location Database

Figure III – 8 shows the SQL Schema including UCON and Location databases. UCON database consists of the following tables: SubObjCont, SubObjAttributes, Rights, Policy Rule, Predicate, ABC, Update, and Mutation.

[9] http://docs.oracle.com

SubObjCont is defined for subjects (s), objects (o) and context. A subject (e.g. human or device) requests to exercise some rights on an object (e.g. a printer or a file). A context could be an environmental variable (i.e. Location). SubObjAttributes represents information that is related to subjects and objects (attributes). A subject and an object could have more than one attributes, either mutable or immutable. Rights table is defined in order to declare the type of rights while Policy Rule table determines the policy rules, according to UCON description in Chapter 2. Predicate table allows the comparisons between a name of an attribute and a specific value. This table is essential for the evaluation of the conditions. ABC table is defined in order to declare the type of ABC model components, such as authorizations (i.e., preAuthorizations, ongoingAuthorization), oBligations (i.e., preoBligations,ongoingoBligations), Conditions (preConditions, onConditions), along with the policy rule and the predicate. Update table keeps the attributes updates (i.e., PreUpdates, onUpdates) that are going to be held, while Mutation table keeps the update type along with the policy id, the update id and the frequency factor, in order to represent the continuity of decision.

In order to extend the functionality of our application adding features related to user's location (i.e. context in SubObjCont table), we have added a Location Database to keep a list of pairs of IP addresses and country prefixes. These prefixes can be easily replaced by city prefixes, as needed.

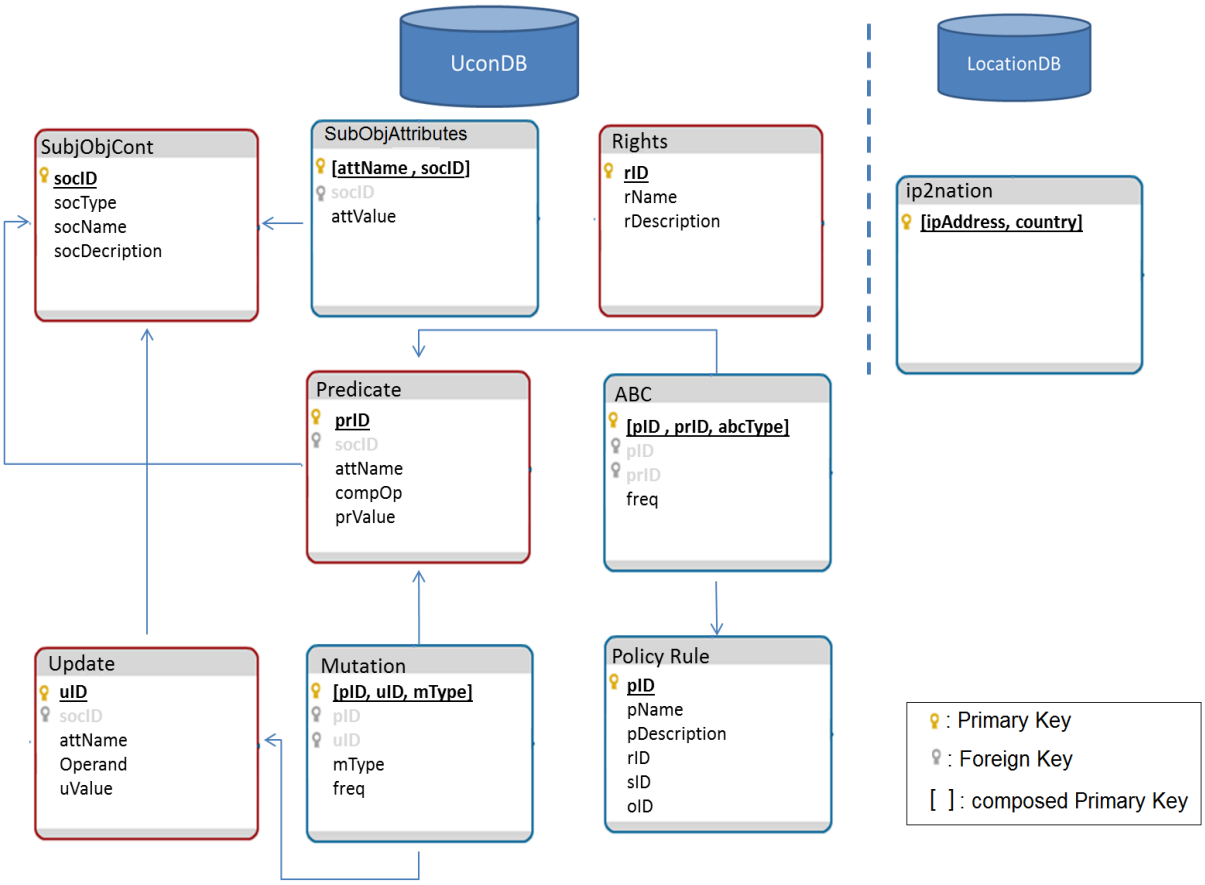

Figure III - 8. SQL Schema for UCON

Example: Alice works in the Computer Science Department as a software developer of a usage control model in a company in France. In order to test her software, she needs some pictures (e.g. image1.jpg) that are located in a specific folder (path: meta/objects/) in the central system of the company. However, she can access these images for reading and editing seven times per day, for five seconds each. Thus, accessing rights, policy rules and conditions must be fulfilled, in order folder access to be achieved.

A sample input for our UCON Database that involves all the tables described in order to implement our scenarios includes:

SubObjCont:

- **Subject**: subject (socType), s1 (socID), Alice (socName), Computer Science Department (socDescription).
- **Object**: object (socType), o1 (socID), "image1.jpg" (socName), "meta/objects/" (socDescription).
- **Context**: context (socType), c1 (socID), Location (socName), "France" (socDescription).

SubObjAttributes:

- **Subject Attributes:** s1 (socID),Department, Specialization, login (attNames) with values ComputerScience, UCONmodel, aliceLogin (attValues), accordingly, as three different entries.
- **Object Attributes:** o1 (socID), accessMilliSeconds (time in milliseconds that the object can be accessed - attName),5000 (attValue) and o1 (socID), accessTime (how many times the object can be accessed - attName), 0 (attValue).

Rights:

- **Accessing rights:** r (rID), read (rName), LoadAnImage (rDescription) and w (rID), write (rName), EditAnImage (rDescription)

PolicyRule:

- **Policies**: p1 (pID), AccessLoad (pName), s1 (sID), 'r' (rID), o1 (oID) and p2 (pID) AccessEdit (pName), s1 (sID), 'w' (rID), o1 (oID). These two policies indicate that s1 could read and write o1.

Predicate:

- **Predicates**:pr1 (prID), s1(socID), Department (attName), '=' (compOp), ComputerScience (prValue). We introduce a comparison operator (compOp) in order to compare a specific subject or object attribute with a specific value

ABC:

- **ABCtype_preConditions:**p1(pID), preCondition (abcType), pr1 (prID), 0 (freq).
- **ABCtype_onConditions:**p1(pID), onCondition (abcType), pr2 (prID), 1000 (freq). The frequency factor (1000) means that the checks of conditions will be done every 1000 milliseconds.

Update:

- **Updates:** u1(uIDs),o1 (socID), accessTime (attName), '+' (Operand), 1 (uValue). We introduce the operand attribute in order to change the uValue accordingly, each time that the objects are accessed.

Mutation:

- **Mutations_preUpdates:** p1(pID), u1 (uID), preUpdate (mType), 0 (freq).
- **Mutations_onUpdates:** p1(pID), u1 (uID), onUpdate (mType), 1000 (freq). The frequency factor (1000) is the amount of millisecond that the mType has to be checked.

Location:

- **LocationCorrelation**: 193.51.25.232 (ipAddress), fr (country). By retrieving the user's IP, we are able to know about her location and also check the Location factor along with other conditions. For instance, as we mentioned before, we have the context c1 where the Location is France. We retrieve the socDescription (i.e. name of the Location) to compare it with subject’s country. If the user's location matches to socDescription then the preConditions related to context are satisfied and then access is granted.

3.1.2.2 Functions of the UCON Engine

In this section, we illustrated all the functions that we have developed in order to implement the $UCON_{ABC}$ model as an Engine that regulates the access on objects. Figure III - 9 depicts the Data Sharing Application that includes the graphical interface (application part) and the UCON Engine (UCON engine part), which is a set of functions that regulates the access related to specific objects from specific subjects.

More precisely, the Data Sharing Application has been developed in four parts. The first part consists of the graphical user interface (untrusted part). The second part includes the manager of the UCON Database which contains all the appropriate functions in order to initialize the Database, create the essential tables (see Section1.2) and insert values into these tables (Database Part). Graphical user interfaces tend to be complex and vulnerable to spoofing (e.g. key loggers, fake dialog windows); hence there is a need of a trusted path between the user and the application interface itself. Thus, the third part consists of the functions which check the conditions (preConditions, ongoingConditions) and if the conditions are met, then the UCON Engine gives the permissions to a subject for accessing a specific object (trusted part). It is worth mentioning that the conditions are checked not only before the access is granted but also during the access. If the conditions are no longer satisfied, then the UCON engine terminates the access. The fourth part includes the Location Database manager in order to initialize the Location Database and create the appropriate tables (see Section 1.2.2).

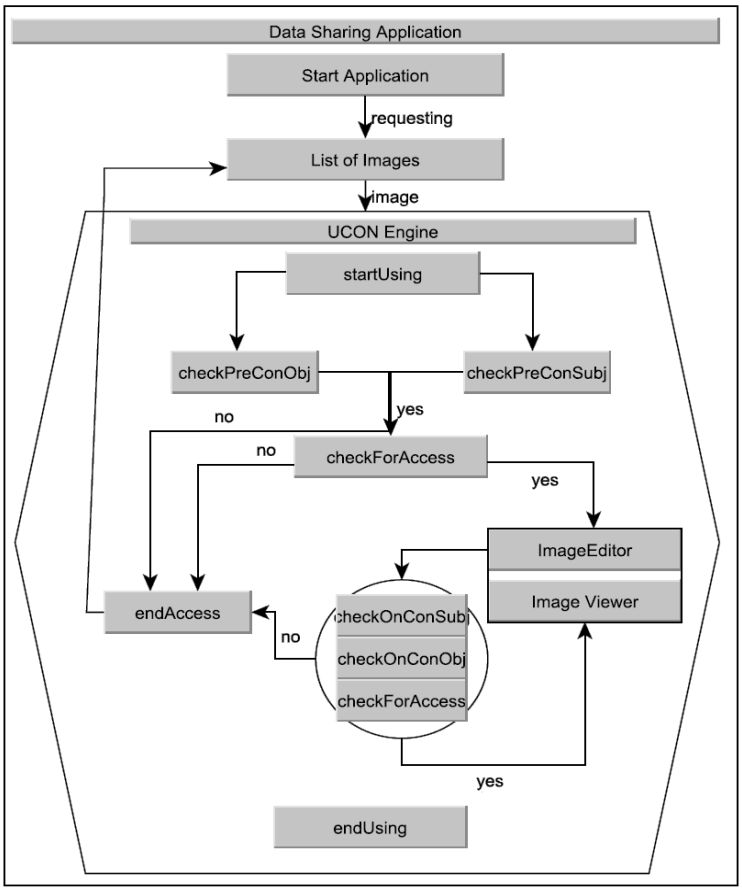

Figure III –9. Data Sharing Application

3.1.2.3 Execution of Data Sharing Application

First, we should initialize the UCON Database. We have developed functions that attempt to connect to the Derby Database[10] and create the tables. Second, we initialize the Location Database by executing the appropriate functions that connect to the Database and create the location table.

Having initialized our Databases, we execute the application part. In order to login to the application, the user has to provide his/her username in the appropriate field and then to click the Login button (see Figure III - 10). Once a user logs in the Data Sharing

[10] Apache Derby Database is an open source relational database implemented entirely in Java and available under the Apache License, Version 2.0. (https://db.apache.org/derby/)

Application (application part), he is directed to a list of images (application part) which is restricted and can view some or all the images under some specific conditions. In order to view an image, he has to click on the image and then press the Load or Edit button. Then, the UCON engine is called in order to enable the usage control of images in our application. The function startUsing(String subject, String object, String mode) is called. This function enables the UCON engine. The arguments of this function include the subject's and object's name and the mode that the subject has chosen to access the object ('load' or 'edit').

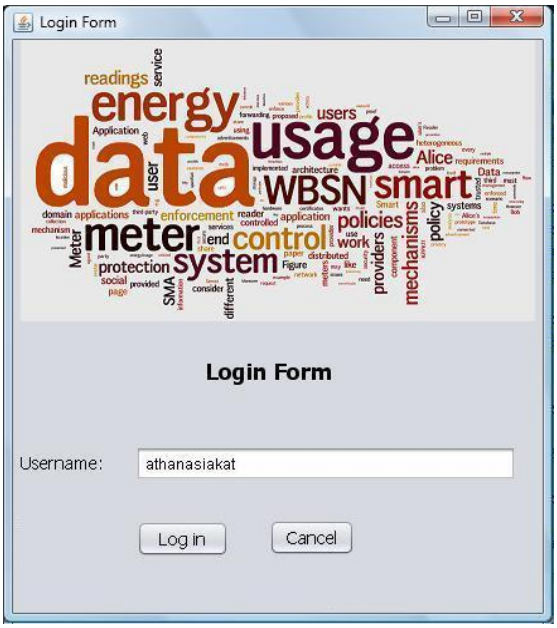

Figure III – 10. Login Page

In order to check the preConditions, the application calls the functions checkPreConSubj(String subject, String object, String condType, String mode), checkPreConObj(String subject, String object, String condType, String mode), and checkPreConCon(String context). The first function checks the preConditions of a subject. The second function checks the preConditions of an object. The third function checks the preConditions of the context. If preConditions are met successfully, then the function checkForAccess(String subject, String object, String context, String mode) is called, in order to check if the specific right ('load'/'edit') is allowed for the pair (subject, object). If so, then s/he will be redirected to the Image Viewer or Image Editor (see Figure III - 11 and Figure III - 12).

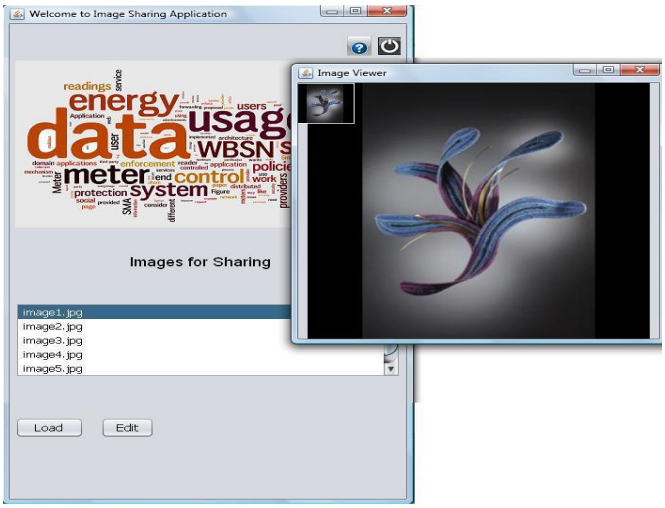

Figure III - 11. Load Image

Then, the ongoingConditions are enforced to ensure the control during the access. The startUsing function calls the functions checkOnConSubj(String subject, String object, String condType, String mode) and checkOnConObj(String subject, String object, String condType, String mode) to check the ongoingConditions of the subject and object, respectively. If the ongoingConditions are met then the subject could access the object.

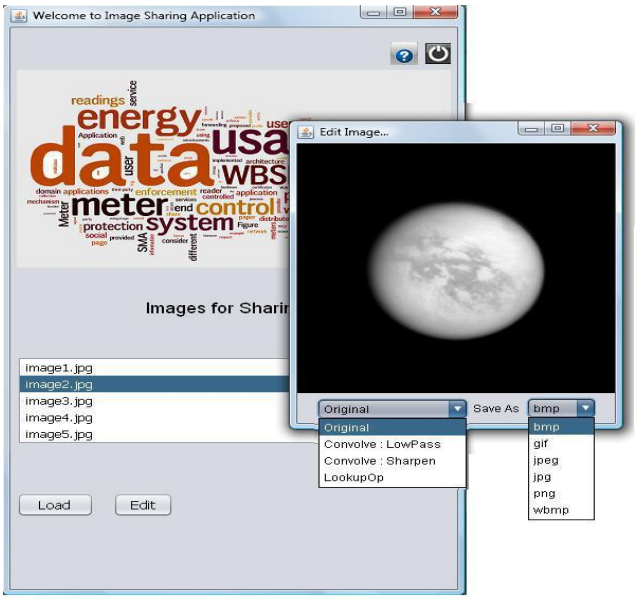

Figure III - 12.Edit Image

It is worth to mention that both Image Viewer and Image Editor have been implemented and called from the part of UCON Engine (trusted part). A notification message may be displayed if the user has not got the permissions for the selected image (see Figure III - 13).

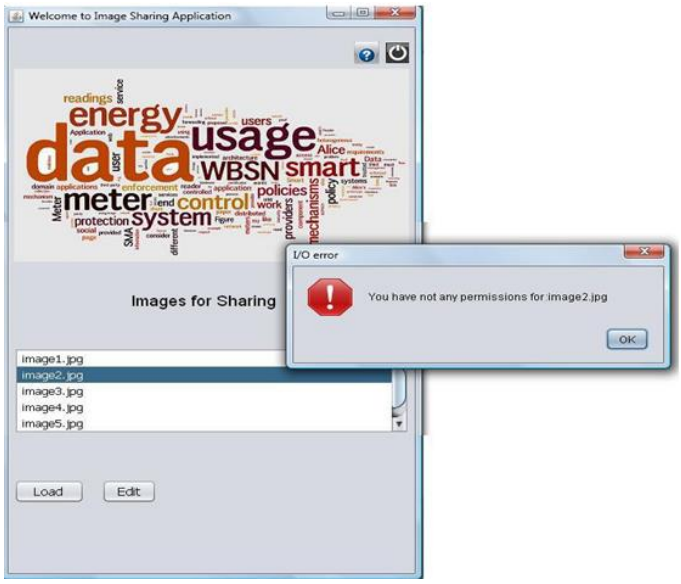

Figure III – 13. No permission for Loading/Editing Image

3.1.2.4 Scenarios

In this section, we present two scenarios that allow loading/editing an image. Let us assume that Alice (subject) works at the Computer Science Department (attribute) and she is responsible for designing the $UCON_{ABC}$ model (subject attribute). She has been given a username aliceLogin (attribute) from her Department. Furthermore, her Department has a folder that contains images (objects) which can be viewed or edited under some specific conditions (preConditions or/and ongoingConditions) that have to be fulfilled.

Alice is logged in the application by providing her username and directed to the list of images. When Alice clicked on the image and then pressed the Load or Edit button, the UCON engine (ucon engine part) is enforced (see Figure III - 9) in order to enable the

usage control of images and check if this user has the appropriate permissions for accessing ('load' or ‘edit’) this image.

Alice (subject) has to belong to the Computer Science Department (preCon) in order to access (right: write) "image2.jpg"(object), twice (mutability) from this folder while and she has additionally to be in France (contextual information) in order to have the permission for accessing (right : read) "image1.jpg" (object), 7 times (mutability) for 5 seconds (onCon) each.

Scenario # 1: Load an Image

The UCON Engine starts (startUsing("aliceLogin", "image1.jpg","France", "load")) by checking the preConditions for the subject (checkPreConSubj("aliceLogin", "image1.jpg", "preCon", "load")), the preConditions for the object (checkPre ConObj("aliceLogin", "image1.jpg", "preCon", "load")) and the preConditions for the context (checkPreConCon("France")).

More precisely, in our case, the UCON Engine checks if aliceLogin belongs to Computer Science Department (see the Tables in Section 1.2.1) and if the access time is elapsed. It also checks if the location is France (see Table III - 10) by retrieving the IP address of aliceLogin and calling the function returnCountry("193.51.25.232", "Alice-PC", stmt, "fr") that returns the name of the country. If the preConditions are met, then it checks if she has permissions for exercising the load operation (right:'read') (checkForAccess("aliceLogin", "image1.jpg", "load")). If so, the attributes which are related to accessTime (accessTime = accessTime + 1), as well as the variable which represents the seconds of each access are changed (mutability), otherwise she is redirected to the list of images.

Then, the ongoingConditions are enforced (checkOnConSubj("aliceLogin", "image1.jpg", "load"), checkOnConObj("aliceLogin", "image1.jpg", "load")), in order to check if the conditions are met or not. If she has not the appropriate permission for loading the image or when conditions are not met anymore (i.e. the available seconds for loading this image are elapsed) then she cannot exercise any rights to the image (see Figure III - 13) and she is redirected to the list of images again.

Scenario # 2: Edit an Image

The UCON Engine starts (startUsing("aliceLogin", "image2.jpg", "edit")) by checking the preConditions for the subject and object (checkPreConSubj("aliceLogin", "image2.jpg","preCon", "edit"), checkPreConObj("aliceLogin", "image2.jpg", "preCon", "edit").

More precisely, in our case, the UCON Engine checks if aliceLogin belongs to Computer Science Department (see the Tables in Section 1.2.1) and if the access time is elapsed. If the preConditions are met, then it checks if she has permissions for exercising the edit operation (right:'write') (checkForAccess("aliceLogin", "image2.jpg", "edit")). If so, the attribute which is related to accessTime (accessTime = accessTime + 1) (mutability) is changed, otherwise she is redirected to the list of images. Then, the ongoingConditions are enforced (checkOnConSubj("aliceLogin", "image2.jpg", "edit"), checkOnConObj("aliceLogin", "image2.jpg", "edit")), in order to check during the access if the conditions are met or not. If she has not the appropriate permission for editing the image or when conditions are not met anymore then she cannot exercise any rights to the image (see Figure III - 13) and she is redirected to the list of images again.

3.1.3 Conclusion

In this section, building on the state of the art in access and usage control, we have presented in detail, two implementations of the UCON model: the former one includes a usage control engine that could be embedded within a system, while the latter one includes a data sharing application in which we have embedded the UCON Engine. In this implementation, we used a relational database in order to extend and simplify the first implementation, providing a new perspective on the implementation of this model. Despite the powerful features that this model offers, there are still some cases that are not covered such as the case that data sharing involves several users or even groups of users, where the data types as well as sensitivity levels can be vary. Thus, in the next Section, we introduce DatShA, a data sharing algebra complementary to the UCON model giving users the tools to define their own access control plans over their data.

3.2 DatShA: Data Sharing Algebra

As already mentioned in Chapter 2, online social networks (OSN) are one of the most successful applications that have been created this last decade. Central to these applications is the problem of sharing data, such as texts, photos, geo-location, etc. In most cases, this data is private, and thus is only shared with “friends”, a loose concept. Some OSN, such as Google+ let you define circles in order to categorize your friends: friends, close friends, acquaintances, etc. Data can then be shared on finer grain using these circles. However, there is no automatic way to control the simultaneous sharing of data to several circles, with different data precision granularities, such as in the following scenario: Alice wants to share a set of photos with her family, photos with no metadata with her close friends, photos without faces (and without metadata) in a reduced definition with her acquaintances, and does not want to share anything with anyone else.

In this section, we will show how the use of a data sharing algebra to write a variety of access control plans (ACP) can overcome these current limitations of OSN access control. Moreover, by using an algebra, it becomes simple to modify, compose, and share these ACPs. Thus, less advanced users can easily reuse ACPs shared on a marketplace by more experienced users. A prototype of the DatShA system has been implemented using XQuery 3.0[11] [Eisenberg13] and is briefly described.

3.2.1 Overview of DatShA

In current OSNs, users have on one side vast quantities of personal data, and on the other side numerous “friends” with whom they wish to share (or sometimes hide) this data. In the current systems, it is not obvious how to share a specific piece of data while modifying it (e.g. changing its precision or removing information) depending on the target with whom it is shared.

Consider the examples mentioned above. The ACP related to Alice’s close friends should transform a set of photos to another set where metadata is removed. This could be done by simply specifying a regular expression to identify images files to be shared

[11] https://www.w3.org/TR/xquery-30/

(FileSearchoperator – see Figure III – 15.b), “type” of these files to images (PathToImage operator – see Figure III – 15.b), then remove metadata (RemoveMeta operator). For Alice’s acquaintances, other operators could be invoked: ExtractFaces, ExtractMeta, Select and ReduceDefinition operators.

Thus the objective of DatShA is to provide the infrastructure and an extensible set of generic operators to describe how users would prefer to process their data before sharing it. The operators must be able to be combined on any sort of (semi-structured) data to form an algebra. Finally, ACP may include user-dependent data (e.g., contact files) such that it can also compute the set of users with whom the data is shared, thus linking a plan with its grantee.

3.2.2 Background and Related Works

Access Control

Many different access control models exist, such as DAC, MAC, or R-BAC, as we discussed in Chapter 2. Many works exist on enforcing such models in OSN [Carminati09]. We adopt a complementary approach: the goal of DatShA can be seen as helping the user to write complex views of her data, on which she can then apply any existing Access Control model (most often, DAC or RBAC) and Usage Control one ($UCON_{ABC}$).

Data Sharing on OSN

Current works on secure data sharing in OSNs consider various problems such as securing communications, i.e. how to securely share data, once access control has been checked [Qinlong14], or how to write access control policies over data concerning several users [Hu13].

XQuery 3.0

XQuery 3.0. [Eisenberg13] is not only a declarative query language, it is also Turing complete. Rather than using a traditional language such as Java or C, we have chosen to use XQuery and XQuery Update Facility 3.0. Indeed, evaluating an ACP is done through modifications of a structured document (that we chose to encode in XML).

Generic operators can be completed by snippets of XPath or XQuery code referring to this data structure, which are directly evaluated by the DatShA system.

3.2.3. Data Sharing Algebra (DatShA)

3.2.3.1 General principle

An ACP is seen as a set of sequences of (polymorphic) operators, serialized as an XML file (see Figure III - 14). It takes as input an XML file containing or referencing private sensitive data and produces an XML file containing or referencing data that can be shared or published (See Figure III – 15.a). Users or sets of users (such as Google+ circles) can be given access rights both on atomic data, and on ACPs. As with traditional access control through views, when access rights are given on an ACP, the data accessed during the process is done with the rights of the grantor. For example, if Alice grants Bob the right to view the country where she is in, which is computed using her precise GPS coordinates, the execution of the ACP will use Alice's rights, but only return to Bob the final result.

3.2.3.2 Java/XQuery Implementation

In this section, we are going to discuss the implemented operators in details.

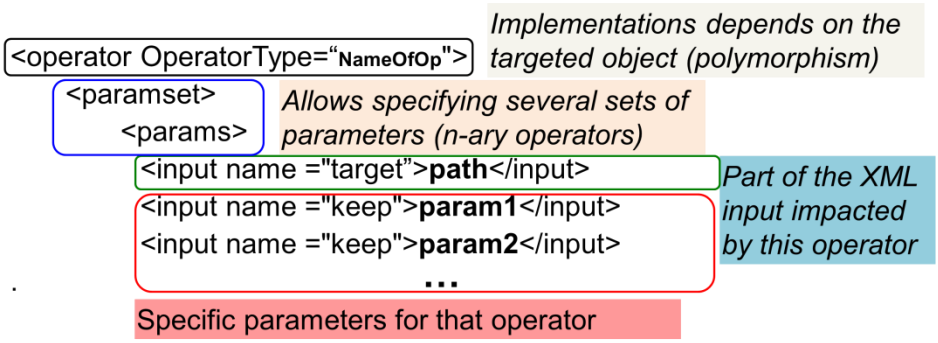

Figure III - 14. General Definition of an Operator

Figure III -14 shows the general form of an operator, as it is defined in an XML file. The definition of each operator includes its name and a set of parameters that is different for each operator and is related to its functionality.

We distinguish two different types of operators:

- **Ad-hoc Operators:** An API that implements file-oriented operators has been designed in Java. In order to implement this type of operators, we have used data structures such as ArrayLists and NodeLists in Java. Each operator that falls into this type has its own function and provides the corresponding functionality. An example of such operators could be the following:

  - **fileSearch**: This operator takes as an input two strings that represent a directory path and the full path of the output xml file and returns an xml file storing the results (a set of objects of the same type).

fileSearch_op

Input: String fileSpec, String xmlOut

Output: String xmlOut

```
public void fileSearch_op(String xmlIn,String target,String xmlOut){
   String xExpr =Utilities_Nodes.getPathValue (xmlIn, target);
   //path list ( fileSpec in the form: fullpath/*.ext)

   ArrayList < String> setOfPaths = Utilities.getSetOfFiles(xExpr);
   //create File2.xml :output of this operator
   //create the document
   Document dom = Utilities_Nodes.createDocument();
   //retrieve the type(e.g. image,doc,audio,video)& add it as a node
      String filespec =
      Utilities.returnFileType(setOfPaths.get(0).split("\\.")[1]);
   Utilities_Nodes.createDOMTreeSet(dom,  setOfPaths);
   Utilities.printToFile(dom,xmlOut);
```

Functionality:

- First, we get a set of objects of the same type according to the filespec (input) and we store it in a List.
- Then, we retrieve the type of this set of objects in order to create the corresponding node in the output xml file.
- Finally, we create the xml file containing this list.

The output of this operator is the following (File2.xml):

```
<?xml version="1.0" encoding="UTF-8"?>
   <set>
   <path>samples/image1.jpg</path>
   <path>samples/image2.jpg</path>
   <path>samples/image3.jpg</path>
</set>
```

- **XQuery Operators:** This type of operators has been developed in XQuery. Each operator that falls into this category can be expressed in few lines of XQuery; providing the corresponding functionality. An example of XQuery operator includes:

pathToImage_op

```
xquery version "3.0";
for $r in doc("File2.xml")//path
return
    update replace $r with
        <image>
            {$r}
        </image>
```

3.2.3.3 Sharing ACPs through a marketplace

Operators and ACPs can be published on a “marketplace”, and described by a short text explaining their goal. They can be downloaded by users in order to fine tune their data sharing policies. Thus, it is possible, even for non-expert users to apply complex access control policies, by combining existing operators or using existing policies. Search, recommendation, or ranking of ACP or operators based on their level of intrusiveness or their usability is possible within the marketplace. The only complexity is to link groups of users to their ACPs, but as the data shared is defined intentionally rather than extensionally, we believe this is much easier to do than with current privacy settings in OSN.

3.2.3.4 ACP Example

We propose the following example which illustrates well DatShA potential : Alice wants to participate in a survey to determine the most photographed place on Earth, which can be done by computing a “fuzzy” location of all her photos, where the “fuzzy” location is defined by GPS coordinate and an error bar e.g. X=45.23+/-0.01 Y=27.67+/-0.01. Note that this error bar could also be a function of the density of photos in a given area.
We present, in Figure III - 15, the corresponding ACP.

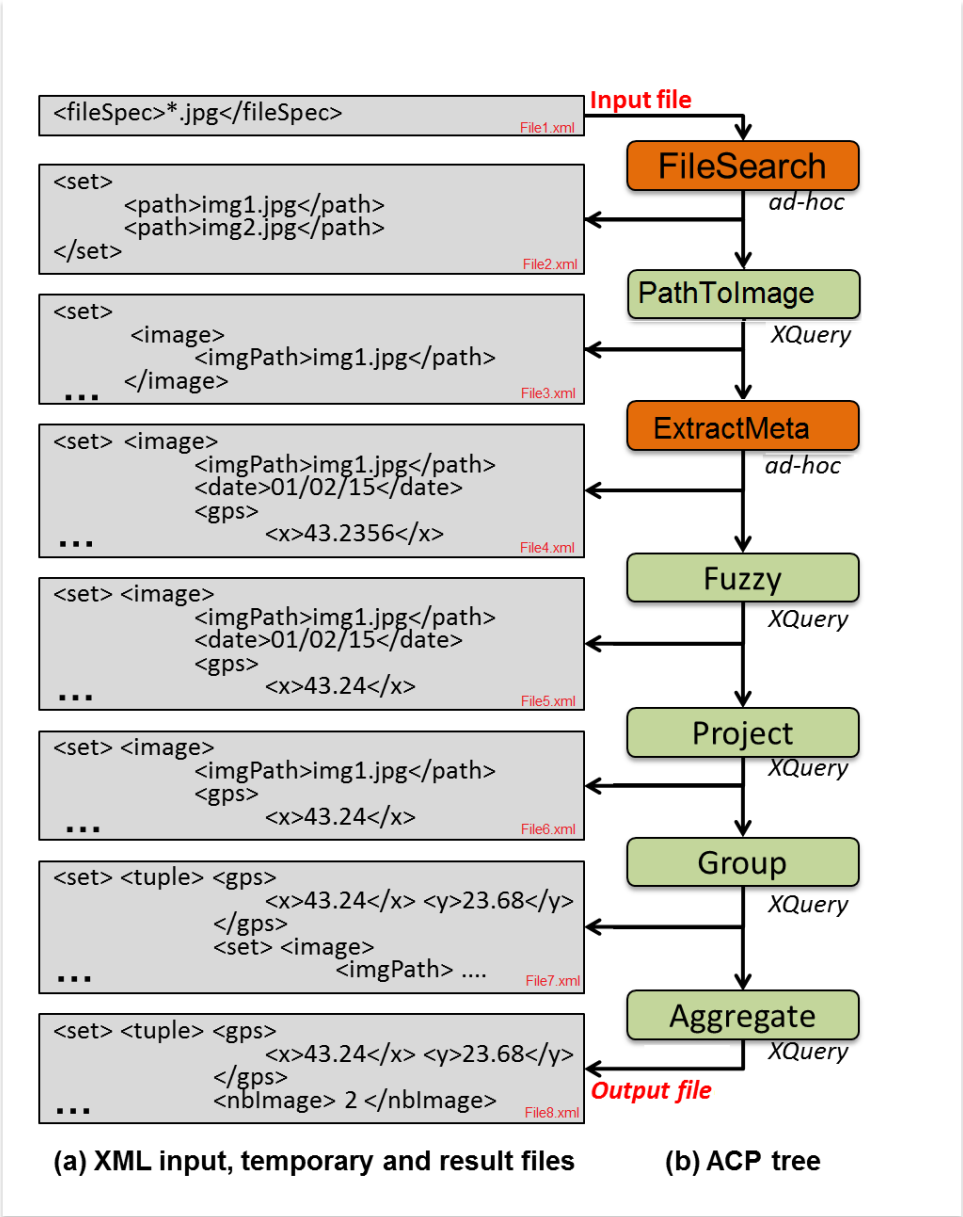

Figure III – 15. Detailed Example of an ACP

This ACP can be written as a sequence of operators. Each operator takes as input an XML file, and produces as output an XML file. It is possible to type-check the ACP at compile time, given that the operators are typed. The sequence is the following: for every image in the file path given in the input file, metadata of the image is extracted, and an operator to reduce the precision of the GPS coordinates is executed. All metadata apart from the blurred GPS coordinates is removed; pictures are grouped together by fuzzy GPS location, and then counted.

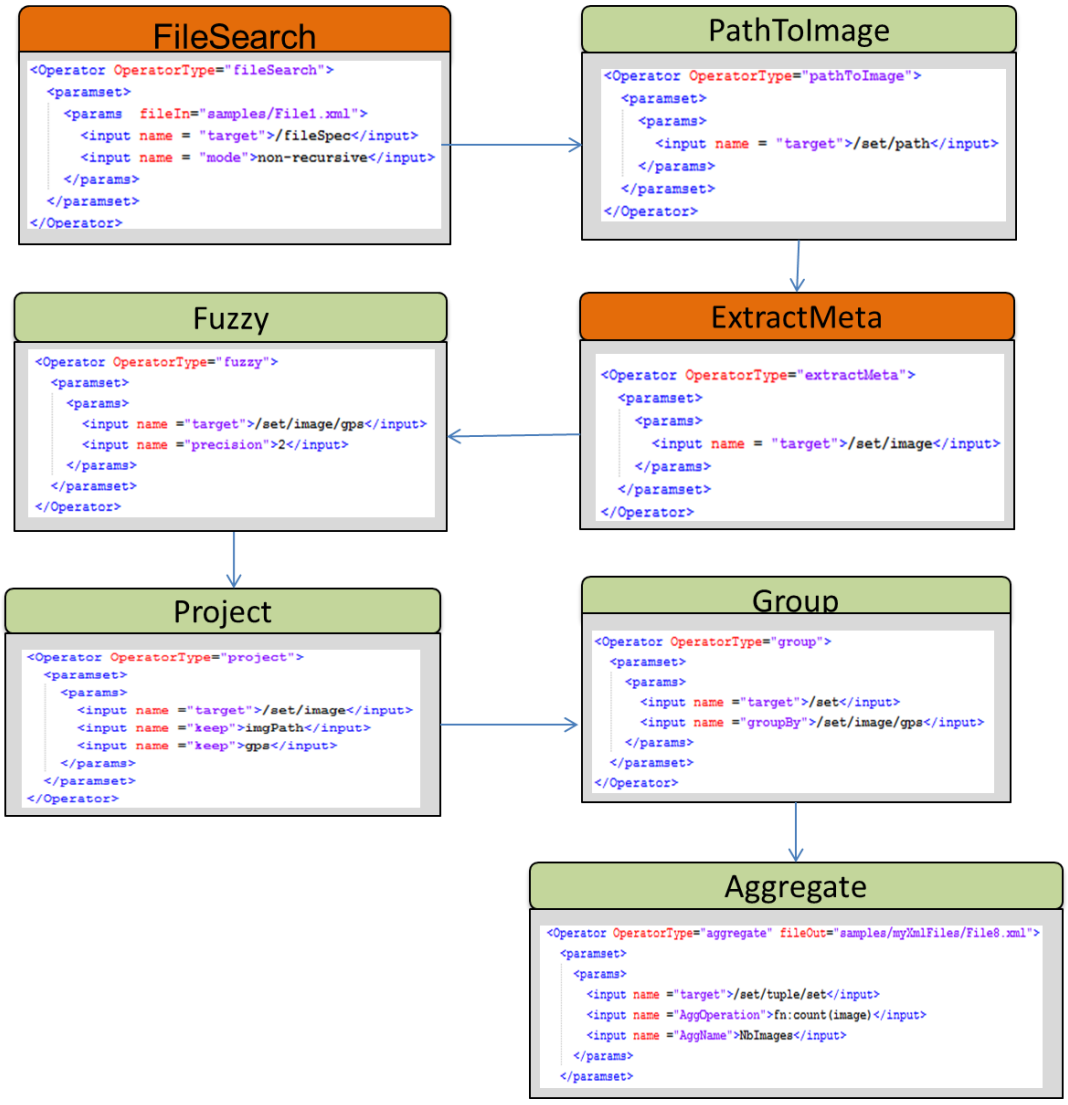

Figure III – 16. Definition of ACP (Linear sequence of operators) (ACP.xml)

In the rest of this section, we present the operators used in this example, in detail.

fileSearch: This operator replaces a <fileSpec/> with jokers in a set of file paths looking in the directory indicated by input "target". This could be done either recursively or non-recursively. In detail, every <fileSpec>will be replaced by a set of paths. In our case, there is only a single fileSpec; hence leading to a single set of paths.
The input xml file (File1.xml) is the following:

```
<fileSpec>samples/*.jpg</fileSpec>
```

The definition of filesearch operator is the following:

```
<Operator OperatorType="fileSearch">
<paramset>
<params  fileIn="samples/myXmlFiles/File1.xml">
         <input name = "target">/fileSpec</input>
   <input name = "mode">non-recursive</input>
</params>
</paramset>
</Operator>
```

Once the operator is applied to File1.xml, the output xml file (File2.xml) will have the following format:

```
   <?xml version="1.0" encoding="UTF-8"?>
<set>
   <path>samples/image1.jpg</path>
   <path>samples/image2.jpg</path>
   <path>samples/image3.jpg</path>
   </set>
```

pathToImage: This operator replaces every occurrence of an image path by an image (type: image). An image is at least an `<imgPath>`that corresponds to an "image" file.

We define pathToImage operator as follow:

```
<Operator OperatorType="pathToImage">
<paramset>
<params>
         <input name = "target">/set/path</input>
</params>
</paramset>
</Operator>
```

The generated XQuery for this operator is the following:

```
xquery version "3.0";
for $r in doc("File2.xml")//path
return
    update replace $r with
        <image>
            {$r}
        </image>
```

The generated output xml file (File3.xml) after applying this operator to File2.xml will have the following format:

```
<set>
<image>
<imgPath>samples/image1.jpg</imgPath>
</image>
<image>
<imgPath>samples/image2.jpg</imgPath>
</image>
<image>
<imgPath>samples/image3.jpg</imgPath>
</image>
</set>
```

extractMeta: This operator replaces every occurrence of an image by the same image (every field is copied); including the metadata that will be extracted from the actual file. Java libraries are used in order to achieve this metadata extraction.

The definition of extractMeta operator includes:

```
<Operator OperatorType="extractMeta">
<paramset>
<params>
   <input name = "target">/set/image</input>
</params>
</paramset>
</Operator>
```

Once this operator is applied to File3.xml, the output xml file (File4.xml) will have the following format:

```
<set>
<image>
   <path>samples/image1.jpg</path>
   <date>2015/08/20</date>
   <gps>
   <X>66.0204</X>
   <Y>62.2302</Y>
   </gps>
   </image>
   <image>
   <path>samples/image2.jpg</path>
   <date>2015/12/25</date>
   <gps>
   <X>29.1708</X>
   <Y>26.1811</Y>
   </gps>
   </image>
      <image>
   <path>samples/image3.jpg</path>
   <date>2016/01/01</date>
   <gps>
   <X>30.1986</X>
   <Y>26.1989</Y>
   </gps>
   </image>
   </set>
```

fuzzy: is an operator that can be applied to many types. Its global behavior is to replace any occurrence of the target by fuzzy values; the precision being informed by the “precision” input, which can be an XPath.

The definition of fuzzy operator is the following:

```
<Operator OperatorType="fuzzy">
<paramset>
<params>
         <input name ="target">/set/image/gps/*</input>
   <input name ="precision">2</input>
</params>
</paramset>
</Operator>
```

In this case, fuzzy operator is applied to geo-location data (i.e. gps data); to X (float) and Y (float), reducing the precision. Once, this operator is applied to File4.xml, the output xml file (File5.xml) will have the following format:

```
<set>
<image>
   <path>samples/image1.jpg</path>
   <date>2015/08/20</date>
   <gps>
   <X>66.02</X>
   <Y>62.23</Y>
   </gps>
   </image>
   <image>
   <path>samples/image2.jpg</path>
   <date>2015/12/25</date>
   <gps>
   <X>29.17</X>
   <Y>26.18</Y>
   </gps>
   </image>
      <image>
   <path>samples/image3.jpg</path>
   <date>2016/01/01</date>
   <gps>
   <X>30.19</X>
   <Y>26.19</Y>
   </gps>
   </image>
   </set>
```

project: this operator is used like the relational algebra π operator; replacing the target subtree by the same subtree in which it keeps only either the elements or subtrees that are mentioned in the `<keep>` parameters.

We define project operator as follow:

```
<Operator OperatorType="project">
<paramset>
<params>
   <input name ="target">/set/image</input>
   <input name ="keep">imgPath</input>
   <input name ="keep">gps</input>
   </params>
</paramset>
</Operator>
```

The generated XQuery for this operator is the following:

```
xquery version "3.0";
<set>
{
   let $doc := .
   for $v in in doc("File5.xml")//image,
   return
      update replace $r with
<image>
            {$v/imgPath $v/date, $v/gps}
</image>
}
</set>
```

Once this operator is applied to File5.xml, the output xml file (File6.xml) will have the following format:

```
<set>
<image>
   <path>samples/image1.jpg</path>
   <date>2015/08/20</date>
   <gps>
   <X>66.02</X>
   <Y>62.23</Y>
   </gps>
   </image>
   <image>
   <path>samples/image2.jpg</path>
   <date>2015/12/25</date>
   <gps>
   <X>29.17</X>
   <Y>26.18</Y>
   </gps>
   </image>
      <image>
   <path>samples/image3.jpg</path>
   <date>2016/01/01</date>
   <gps>
   <X>30.19</X>
   <Y>26.19</Y>
   </gps>
   </image>
   </set>
```

Group: replaces the "target" subtree by a restructured one which must be a set. It constructs a <set> of <tuple>s, each containing n+1 elements (where n is the amount of "groupBy" elements in the operator specification, in this example, n = 1). The last element of the tuple is a set of elements that share the same value of groupBy (here a set of images having the same GPS value).

We define group operator as follow:

```
<Operator OperatorType="group">
<paramset>
<params>
<input name ="target">/set</input>
   <input name ="groupBy">/set/image/gps</input>
</params>
</paramset>
</Operator>
```

The output xml file (File7.xml) that will be generated after applying this operator to File6.xml will be the following:

```
<set>
<tuple>
   <gps>
   <X>66.02</X>
   <Y>62.23</Y>
   </gps>
   <set>
   <image>
   <path>samples/image1.jpg</path>
   </image>
   <image>
   <path>samples/image2.jpg</path>
   </image>
         <image>
   <path>samples/image3.jpg</path>
   </image>
   </set>
   </tuple>
   </set>
```

aggregate : The aggregate operator replaces a set of elements ("target" input) by an aggregate value having the “AggName” name and applying the “AggOperation”, which in this case is the XQuery function fn:count().

We define aggregate operator as follow:

```
<Operator OperatorType="aggregate"
fileOut="samples/myXmlFiles/File8.xml">
<paramset>
<params>
   <input name ="target">/set/tuple/set</input>
   <input name ="AggOperation">fn:count(image)</input>
<input name ="AggName">NbImages</input>
</params>
</paramset>
</Operator>
```

Thus, once this operator is applied to File7.xml, the output xml file (File8.xml) will have the following format:

```
   <set>
<tuple>
   <gps>
               <X>66.02</X>
               <Y>62.23</Y>
   </gps>
<NbImages> 3 </NbImages>
   </tuple>
   </set>
```

3.2.4 Conclusion

In this section, we introduced DatShA, a data sharing algebra that can be used to create ACPs to manage access control to individuals’ data. The full power of DatShA appears when users start sharing ACPs between each other, either by simply reusing an ACP written by another user, or by integrating such an ACP into a more complex one. Indeed, any ACP can be encapsulated as a DatShA operator. Creating an online marketplace, and testing its usability and adoptability by real users would be the next step of this work, which we leave as a future research direction.

Chapter IV
Experimental Work

In Chapter 3, we described in detail, two implementations of a usage control model named UCON model, as well as we introduced DatShA, a data sharing algebra that provides users with the tools to define their own access control plans over their personal data. This chapter details the experimental work that has been carried out in the context of data sharing and data privacy protection. This work promotes secure data exchange between various users and protection of data privacy by proposing two different infrastructures. First, we introduce a privacy-by-design data sharing infrastructure that provides the individuals with the necessary software modules allowing them to store and share their personal data securely, using a secure device (SPT). This data sharing platform is currently used in the context of privacy laboratory courses where the students are able to interact and develop the existing infrastructure through programming activities. The students are divided into groups of 3 to 4 people, each group being responsible for a specific system module. Second, we describe an experimentation using both a secure and a vulnerable approach of a system dedicated to Sensitive Questionnaire Surveys. The first approach involves SPTs where both the questionnaire survey and the participants' answers are stored securely while the second approach involves a central server (MySQL DBMS server) to which participants will be connected and which will store their answers to the survey. Our aim is to examine to which extent user behavior is altered by the use of secure hardware in the context of Sensitive Questionnaire Surveys. This work was presented at 6^{e} Atelier sur la Protection de la Vie Privée (APVP'15) [Katsouraki15a] and 31ème Conférence sur la Gestion de Données - Principes, Technologies et Applications – BDA 2015 [Katsouraki15b].

4.1 A SPT based Data Sharing Platform

4.1.1 Introduction

Currently, it is critical to protect personal information since even the most secure servers can be easily attacked. Personal data is the new oil of the Internet and the new currency of the digital world [PersonalData11] and is exploited opaquely by the Internet Majors. In order to return control to the user, the SMIS team (UVSQ-INRIA) has developed a Personal Data Secure Server called PlugDB [Anciaux10, Anciaux13, Allard10], as described in Chapter 2. PlugDB allows individuals to exercise access control over their data, preserving both their availability and sharing possibility.

Considering all the above, this section depicts a data sharing platform that allows individuals to exchange their files securely by using the PlugDB technology and integrating access and usage control, thus protecting their privacy.

4.1.2 Principle of the Secure Data Sharing Platform

The principle of this infrastructure includes individuals that are the owners of a personal digital space surrounded by a secure environment that is called Trusted Cell (TCell). This personal digital space can be located either in their personal computer or encrypted on the Cloud and serves as storage space for all their files (i.e. documents, photos, agenda, invoices). The aforementioned files include data that either has been generated by the individuals themselves (e.g. data from their sensors, smartphone, camera and their personal computer) or has been sent to them (e.g. sent by their employer, bank, Electricity Company). However, personal computers with Internet access face many attacks and information leakage risks. TCells aim to protect this personal digital space since they consist of a secure device (SPT) in which PlugDB is embedded, plugged into the user's terminal which contains an access to the encrypted files.

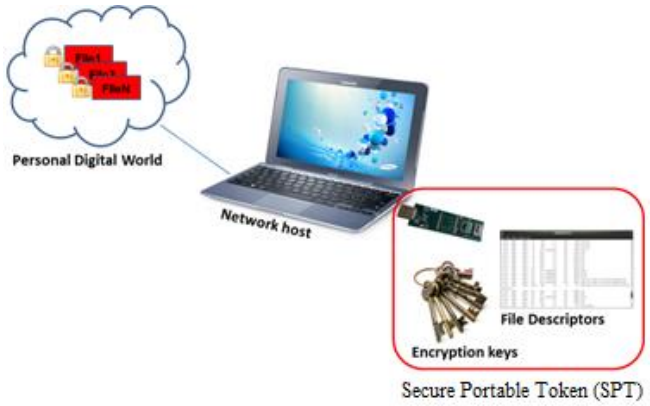

Figure IV - 1. Individual's Digital Space protected by SPT

Two different use cases can be identified:

Single-user

Personal files are stored encrypted in the user's personal digital space located in her personal computer. The keys that are used for encryption are not stored on the computer. The SPT contains a file descriptor table to store the basic files' metadata (metadb) and the encryption keys associated with user's files (one key per file). Once the user is authenticated, his SPT will respond to his requests, such as asking for the key(s) that is (are) associated to files (query on the metadb). For the sake of simplicity, we consider that the keys come from the SPT, the file being decrypted on the terminal. However, security and privacy of the data can be guaranteed since no data access is possible if the SPT is not connected to the terminal and the user is not authenticated.

Multi-user

This platform allows users to exchange their files securely. Consider the following usage example: Alice wants share a file F with Bob, such that only Bob is able to decrypt it and integrate it into his personal digital space. It is worth mentioning that Alice and Bob's SPTs are able to exchange encryption keys. Moreover, the data owner may declare sharing rules (access and usage control rules) that are stored in his SPT and verified by this SPT for each request. The 'subjects' affected by these rules may be either individuals (e.g. Bob) or applications (e.g. Thunderbird). For instance, Alice could specify that file F can be accessible by Bob for a certain period of time (i.e. working hours) and otherwise no decryption is possible.

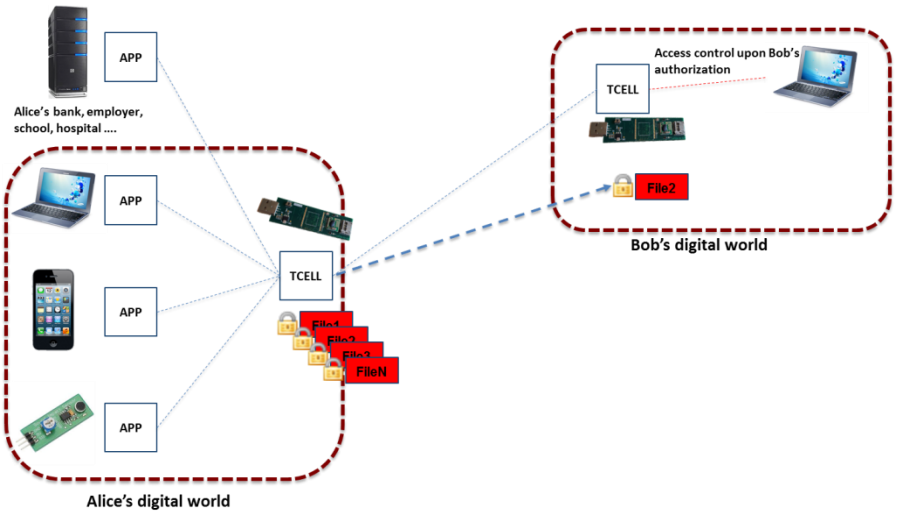

Figure IV - 2 Data Sharing using Secure Portable Tokens (SPT)

The goal was to propose a basic infrastructure to students, such that they can build Privacy-by-Design Applications on top of it. Thus, we tried to keep the platform as simple as possible, giving the students the minimal basic blocks to build their own applications. For instance, the propose infrastructure does not include neither user authentication, nor message integrity and authentication, nor access and usage control features, which are supposed to be added by the students.

4.1.3 Architecture of Data Sharing Platform

In this section, we present the architecture of the aforementioned platform. Privacy has been taken into account in order to design and implement this data sharing platform (privacy-by-design approach).

The data sharing platform consists of the following elements:

- A secure device (SPT) running PlugDB.

- Simulation of SPT (RaspberryPi, SQLite), in order to achieve the functionality of the aforementioned infrastructure without the use of SPT, hence without

hardware security. This simulation was mainly used in the first editions of the teaching lab, because at that time we had not enough SPTs for all the students.

- Application software that implements the communications between the secure device and the unsecure application.

4.1.4 Concepts and Encryption Protocols

The concepts that have been used during the implementation of this platform include:

Identifiers

Each user is identified by a unique identifier **UserGID** (i.e. integer) in the trusted cell ecosystem. Furthermore, each file is identified by a unique identifier **FileID**(i.e. fileName = fileName+filePath) on a given device. Finally, a global unique file identifier **FileGID** is available in the trusted cell ecosystem. This identifier is obtained by concatenating the **UserGID** and the **FileID**(*UserGID*||*FileID).*

Encryption Keys

The user's SPT possesses two keys; a public K_{pub} and a private K_{priv} key to achieve RSA asymmetric encryption (see Figure IV - 3). K_{pub} is disseminated widely and is paired with K_{priv} which is known only to the owner's SPT. In this case, any message can be encrypted for the user, using his public key K_{pub} and can be decrypted only with the user's private key K_{priv}. In our infrastructure, we generate in advance key pairs and then we spread these pairs to each potential user, as well as the public keys of the other users. An API related to the encryption and decryption of the files has been implemented in Java, including the following basic functions:

- GenSymKey: It generates and returns a symmetric encryption key.
- SymEncrypt: It encrypts symmetrically a file with the given encryption key.
- SymDecrypt: It decrypts symmetrically a file using the given encryption key.
- AsymEncrypt: It encrypts asymmetrically a file with the given public key.
- AsymDecrypt: It decrypts asymmetrically a file with the given private key.

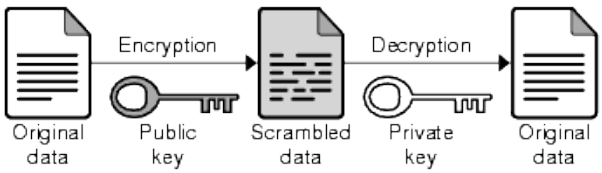

Figure IV - 3. Asymmetric Encryption

A TCell is identified by the IP address of the terminal on which the SPT is connected. The current version of the platform uses a configuration file to inform us about all the IP addresses of the machines that are used.

The files are encrypted using hybrid encryption (see Figure IV - 4). First, we use symmetric encryption (AES) with the same cryptographic keys K_s, for both encryption of plaintext and decryption of ciphertext. When the encrypted file F_{enc} is going to be exchanged between the U_A and U_B, the symmetric key K_s must also be exchanged in order to achieve the file decryption. Thus, U_A encrypts the file F using the symmetric key K_s. Then, the encrypted file F_{enc} is produced. At this point, U_A encrypts K_s with the public key K_B of U_B. U_B will only be able to decrypt K_s, which is then used to decrypt F_{enc} (see Figure IV-4).

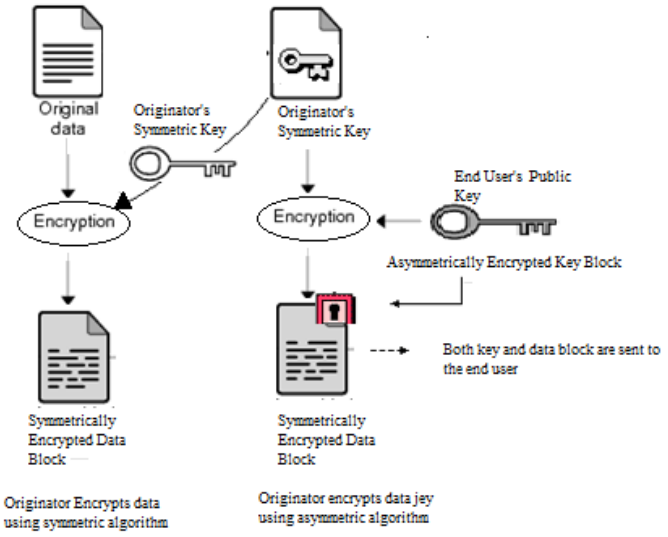

Figure IV - 4. Hybrid Encryption

4.1.5 SPT Platform Architecture and Database Schema

The user's SPT contains a set of metadata organized in relational tables. Figure IV – 5 depicts the architecture of the platform that has been used in this experimental work.

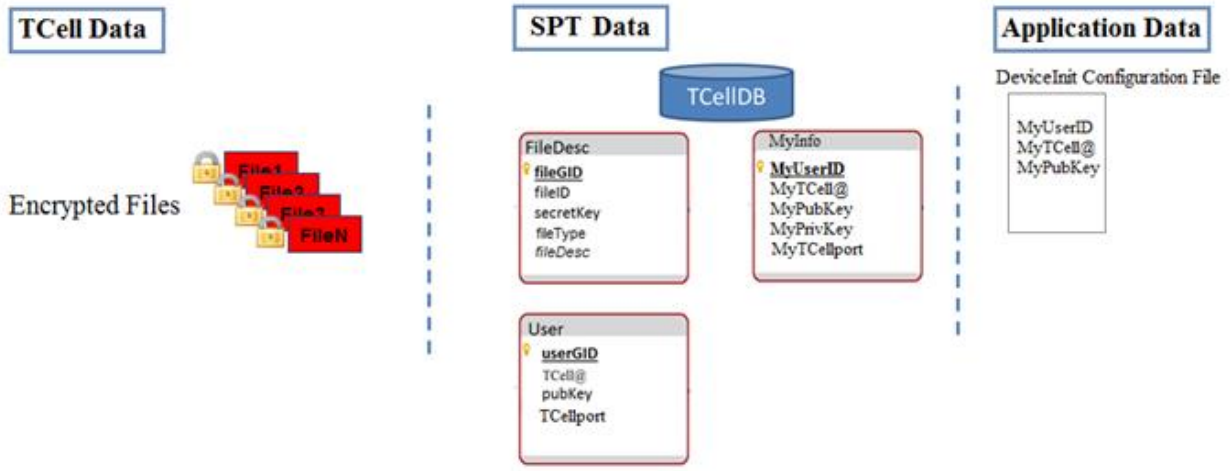

Figure IV - 5. SPT Data Sharing Platform Architecture

Our platform consists of three parts including: Encrypted Files, SPT Secure Portable Token and Application that are installed in individuals' devices (e.g. personal computer).

The user's TCell (terminal) contains his files (TCell files) while the connected SPT's database (TCellDB) includes the following tables:

- **FileDesc:** It contains the files' descriptors that compose the personal digital space of the user. These descriptors include the global file ID (fileGID) that is a concatenation of the userID and the fileID(*UserID||FileID),* identifiers for each file (fileID), the secret key that is needed for the file's decryption (secretKey) and other attributes such as the type of the file (fileType) and its description (fileDesc).

- **User**: It is designed to keep the information needed in order the users to exchange files with other users including: user identifier (UserGID), the TCell

address (TCell@), the public key that is needed for the file's decryption (pubKey) and other attributes such as the port that the SPT is connected.

- **MyInfo**: It is designed to keep the information of SPT's owner including: user's identifier (MyUserID), his TCell's address (myTCell@), his public and private key (MyPubKey and MyPrivKey, accordingly), and attributes such as the port that his SPT is connected (MyTCellport).

Finally, application needs information such as the user identifier (UserID), along with his TCelladdress (MyTCell@) and his public (publicKey). This information is not sensitive and is stored in the DeviceInit files.

4.1.6 Java implemented APIs

An API that allows communications between the users' SPT and the data sharing application has been implemented in Java. It includes the following basic operations:

- **GetFileDesc**: This method has been implemented in order to give the list of files that are stored in the TCell, thus file descriptors in the SPT. As a laboratory work, this module can be enhanced with Access Control.

GetFileDesc Algorithm

Called By: Applications accessing Tcell's files
Input parameters: None
Output parameters: A set of fileGID, Type, Description (from fileDesc)

Build a list based on
Select fileGID, type, Description from fileDesc
Send back the list to the querier

- **ReadFile**: This method has been implemented in order to return to an application the decrypted version of the file. Similarly, we can add Access Control features to this method.

ReadFile Algorithm

Called By: Applications accessing Tcell's files
Input parameters: FileGID: a fileGID
Output parameters: SUCCESS/FAIL

```
Select K, fileID from fileDesc where FileGID = FileGID
If not(ExistLocally(FileID)) then
  Return Fail;
Else
    DecFile = SymDecrypt (FileID, K)
    Send back the DecFile to the querier
    Return SUCCESS;
```

- **StoreFile**: This method has been implemented in order to store an encrypted file in a TCell.

StoreFile Algorithm

Called By: From U's applications to U's Tcell
Input parameters: FileID
Output parameters: SUCCESS/FAIL

```
K = GenSymKey()
EncFile = SymEncrypt (FileID, K)
RemFile = SendFile (EncFile, MyTcell@)
If RemFile == Null then
Return FAIL;
endif
Msg = CreateStoreMsg(RemFile, K) //create a message of type "STORE"
SendMsg(MyPublicKey, MyTcell@, Msg)) //Encrypt & Send msg to TCell@ given
    Return SUCCESS;
```

- **ShareFile**: This method has been implemented in order to transfer an encrypted file from the TCell of U_A to the TCell of U_B or a list of users.

ShareFile Algorithm

Called By: From U_A's Tcell to U_B's Tcell
Input parameters: FileGID: a fileGID
Recipients: list of user GID of the recipients of the sharing
Output parameters: RecipientsOK: list of user GID for which it was successful

```
Select K, fileID from fileDesc where FileGID = FileGID
For each RecipientGID do
    Select Tcell@, PublicKey from user table where UserGID = RecipientGID
    RemFile = SendFile (FileID, Tcell@)
    If RemFile == Null
        Return FAIL;
    Msg =CreateShareMsg(FileGID, RemFile, K)//create a message of type "SHARE"
    //Encrypt & Send msg to TCell@ given
    If SendMsg(PublicKey, Tcell@, Msg) = SUCCESS then
        Add RecipientGID to RecipientsOK
    Endif
EndFor
```

- **ProcessMsg API:** This method has been implemented in order to process a message that is received by a TCell.

ProcessMsg Algorithm

Input parameters: EncMsg: Encrypted message
Output parameters: SUCCESS/FAIL

```
Msg = AsymDecrypt(EncMsg, MyPrivateKey)
If MsgType(Msg) = STORE then
    FileID = GetField(Msg, 1)
    K = GetField(Msg, 2)
    Insert into FileDesc values (MyUserGID||FileID, FileID, K, FileType(FileID), "my file")
    Return SUCCESS;
Elseif If MsgType(Msg) = SHARE then
    FileGID = GetField(Msg, 1)
    FileID = GetField(Msg, 2)
    K = GetField(Msg, 3)
    FileID = GetFile(ExtractFileID(FileGID), Tcell@)
    If FileID = Null then
        Return FAIL;
    endif
    Insert into FileDesc values (FileGID, FileID, K, FileType(FileID), "shared file")
    Return SUCCESS;
endif
```

- **Daemon:** It is always running and is acting when something is transmitted (i.e. message, file). The algorithm that implements the daemon is the following:

Daemon Algorithm

Called By: Running always in Tcell (listening coms and acting when something is sent)
Input parameters: None
Output parameters: None

```
While True do
    If received data is a file then
        FileID = Generate a file name
        store the file locally in FileID
        return FileID
    ElseIf received data is a message then
        processMsg(Msg)
    Endif
EndDo
```

4.1.7 Laboratory Work on SPT based Data Sharing Platform

SPT based Data Sharing Platform enables the development of applications, integrating privacy from the design stage (Privacy-by-Design). It is worth noting that being focused on the protection of personal data the same technological solutions can be applied to any type of sensitive data (e.g. scientific, commercial, industrial). We considered two types of usage for this platform:

- **Educational projects**: Students are supposed to implement Privacy-by-Design applications, based on the proposed platform. Such a project combines the design and the development with a small class of students that can be organized in groups of 4 to 5; each group being responsible for a module such as GUI implementation, file encryption, access control features. One of the groups acts as the architect of the project, being responsible for the functioning of the whole project and guaranteeing the final software integration. Some preliminary sessions were organized such that the students are familiarized with the platform

and learn how to carry out an embedded code. Examples of feasible applications include: secure file sharing, OSN with restricted access, secure portable folders, quantified-self tools.

- **Available to FabLab**: Access to this platform could be available via the FabLab of University of Paris-Saclay, hence students can freely experiment their own innovations.

During the privacy laboratory classes, we were responsible for the installation of the platform and monitoring the students. Our first experience in 2014 at ENSIIE[1] using this platform involved basic SPTs (without the features of the current version such as fingerprint, Bluetooth, speaker). An initial implemented platform in Java (source code) along with the corresponding documentation were provided to the students; allowing them to use SPTs and implement Privacy-by-Design applications on top of it. In the first session, we had used a forge and SVN tool, giving the students the chance to work simultaneously on the source code but also and for us to follow their work.

We had six groups composed by 5 students each. The students were able to choose one of the following topics to develop:

- Design of access control mechanisms (rules to define what is shared with whom, tags association to files in the database).
- Definition of some usage control rules that involve time, location or applications' type (i.e. sticky policies), and propose a way to integrate these rules in the modules of the existing platform.
- Design and development of a graphical interface (GUI) that allows the interaction between the SPT and the individuals' digital space.
- Development of browsers' plugins such that downloaded files metadata will be automatically extracted and stored in the SPT.

[1] www.**ensiie**.fr/

- Development of a recovery server (RS) where all files and messages will be replicated. Additionally, implementation of queries able to retrieve all these messages and files (recovery).
- OSN with restricted access.

In the first edition, one group was dedicated to coordination and integration. In the next edition, each group chose an architect in advance. All the architects were working together in order to:

- Implement the architecture of their module chosen on the top of the initial implemented platform.
- Define the API of each module for all groups
- Manages the different versions (e.g. forge access)
- Coordinate the work and achieve final software integration

4.1.8 Conclusion

This Section depicts the implementation of a SPT based data sharing platform that allows users to store and share personal data securely using a secure device, called Secure Portable Token (SPT). This platform is currently used in privacy laboratory classes by university students, giving them the chance to implement a variety of features such as GUI, access and usage control, and encryption development. The teaching class laboratory has been replicated in 2015 at ENSIEE and UVSQ, and in 2016 at ENSIEE, UVSQ and INSA CVL. Thus, around 150 students have, up to now, used this platform.

In the next Section, we describe another experimental work that was conducted in the context of Sensitive Questionnaires Surveys, involving SPT.

4.2 Sensitive Questionnaire Surveys

4.2.1 Introduction

Surveys and questionnaires are becoming popular, with a wider spectrum of researchers such as social scientists, economists and computer scientists who conduct this kind of study in order to gather data about different human aspects, perceptions, behaviors and attitudes [Andrews03]. However, the more sensitive the questions are, the more skeptical the participants become, leading to either untruthful or lack of response. For instance, when people are asked online questions considering their income, activities and marital status, in the majority of cases, such sensitive questions cause discomfort [Tourangeau07, Ong00, Matsuo04]. In some cases, the participants change the previously provided information, either to prevent an unwanted and shameful exposure, or to avoid any repercussions. This behavior is triggered by the lack of awareness considering access control management and utilization of the provided information. Another point that should be taken into consideration is the fear about potential information leakage of sensitive information that has been already provided online. And this fear is justified: as seen with the PRISM affair, it has become clear that centralizing personal data in a single server managing the questionnaire answers, or in multiple servers controlled by a single actor, introduces a major threat on privacy. Indeed, privacy violations are legion and arise from negligence, attacks and abusive use. No current server-based approach seems capable of closing the gap[2]. In this context, multiple concerns regarding the accuracy of the collected information, as well as the results of conducted surveys, are likely to be raised.

The wide development of secure hardware devices changes the management of sensitive data. Regarding the case of sensitive questionnaire surveys, the answers given could remain in a Secure Portable Token (SPT). Similarly, privacy invasive computations could be done inside the secure hardware. In our work, conducted with researchers from the Réseaux, Territoires, Innovations, Mondialisation (RITM) lab of

[2] http://www.datalossdb.org/

University of Paris Sud, we consider the following questionnaire survey with weighted answers: a questionnaire aiming to suggest careers to students, based on intrusive questions on their likes and dislikes, attitudes at work, etc. These weights are chosen by the economists of RITM and are the same for all users. A particular value representing the suggested career could be calculated in the SPT, using answers and weights[3]. This score will be available to the researchers of the experiment to perform a quantified analysis, while the precise answers will remain private and will be only accessible by the participant.

According to the above considerations, the arising question here is whether individuals would rather disclose their sensitive information by answering a questionnaire survey in a system that provides more tangible security than simply using an online survey web site. It is noticeable that conducting surveys online [Schaniel14, Saaya07] is both a fast and cheap method to gather data, compared to other methods such as paper and face-to-face questionnaires surveys [Doyle05] or questionnaires that need any special equipment to be conducted [Dayan07], although, it includes the risk of receiving no answer from participants concerning sensitive questions.

Our overall objective is to figure out whether people could potentially trust a hardware device (decentralized-storage module) that supports secure storage and management of personal data, to test its adoptability that refers to their willingness to deliver more information supposing that sensitive information is stored on a private device that they own and to test its effectiveness compared to a traditional server, regarding personal data disclosure. Thus, the motivation for the system described in this section emerges from a consideration of the instrumental role of privacy in individuals' lives, and our contribution can be seen as a Privacy Enhancing Technology (PET) to improve surveys.

In order to achieve this goal, we chose to study the impact of the use of this PET in a real field experiment, involving computer scientists who are responsible for setting up the appropriate equipment and implementing an experimental platform (software), and economists, who design the adequate applied-economics protocol. However, organizing and conducting experimentations is not always an easy process, since research

[3] This value is called global score and it is computed as the sum of the detailed scores.

obstacles and difficulties may occur such as equipment and software insufficiencies or participants' difficulty in conceiving clearly the objective of the experimentation. For this reason, a pre-experimentation phase seemed necessary in order to check these particular issues and to conduct experimentations in a correct setting.

Thus, a pre-experimentation was conducted in February 2016 at the University of Paris-Saclay. However, in order to organize such a study, we needed to propose a legitimate goal, and see how participants react. Thus, we were able to test our system in a real course which was organized as a survey taken by the students on their behavior when faced with some situations in a company. The announced objective was to suggest careers to students. This process included a questionnaire survey (see Appendix 1.A) which was submitted to the students. The collected personal data (including sensitive data) has been used to perform skills' evaluation and statistical analysis. With this in mind, we developed both a secure and a vulnerable version of a system that could contribute to perform this pre-experimentation, pointing out the importance of sensitive information protection. In the former case, the questionnaire surveys and participants' answers have been stored in the SPTs. Information that was disclosed included some scores that have been calculated based on answers' weights. In the latter case, a central server of the University of Paris-Saclay was used to keep participants' answers. In order to avoid any influence, the same user interface was used by both versions of the system.

The objective of this pre-experimentation, whose results are still being analyzed by researchers of RITM, is to see if users behave differently when using a SPT rather than a traditional server interface (adoptability) but also to avoid any misunderstandings regarding this secure device. Since this study was based on a simplified applied-economics protocol, it allowed economists to calibrate their experimental protocol, and us, computer scientists, to improve our platform providing better equipment (e.g. tablets versus laptops) and better and more user-friendly graphical interface.

The next experimentation phase is going to take place in November 2016. In the next Sections we are going to describe in detail the principle of the pre-experimentation and the experimentation, our experimental platform and finish with some future works that are in progress.

4.2.2 PlugDB Token for Sensitive Questionnaires

Concerning the questionnaire surveys, the sensitive answers remain inside the SPT, as well as the computations based on weights of each answer. These computations allowed us to perform the participants' profile analysis that could be available to the survey's administrators. This analysis does not reveal any sensitive information beyond the profile's description. Hence, PlugDB (see Figure IV – 6) server could be the answer to sensitive questionnaire surveys.

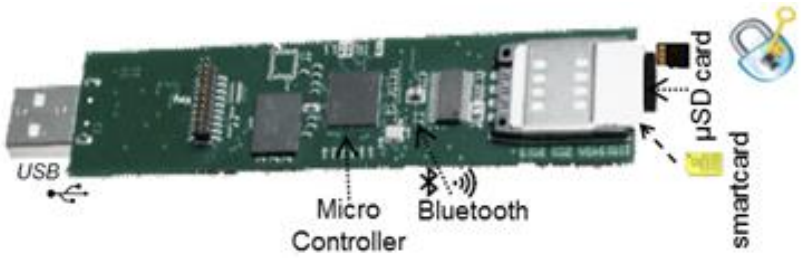

Figure IV - 6. Secure Portable Token (SPT); used in the pre-experimentation

4.2.3 Pre-Experimental Platform and Methodology

In this section, we describe our experimental platform, the pre-experimentation and experimentation protocol and we explained in details the pre-experimentation that took place in February 2016.

4.2.3.1 Pre-Experimental Platform

We developed both a secure and a vulnerable version of a system dedicated to questionnaire surveys. The secure version introduces SPTs in the experimentation process while the vulnerable one involves a central server. MySQL Server was chosen because it is widely used for questionnaires surveys [Schaniel14, Saaya07]. We have

installed MySQL on the server of the University of Paris-Saclay for the needs of our pre-experimentation.

Two groups of students were asked to answer a questionnaire survey related to job seeking, using our platform. The questionnaire suggested eight situations related to competency identification. Each of the answers was assigned with two numbers related to the answer's privacy (privacy score) and the impact of the answer on the user's profile computation (profile score). The privacy score was a positive integer revealing the sensitivity of the answer (the highest, the more sensitive). Privacy and profile score allowed us to provide a profile description to participants.

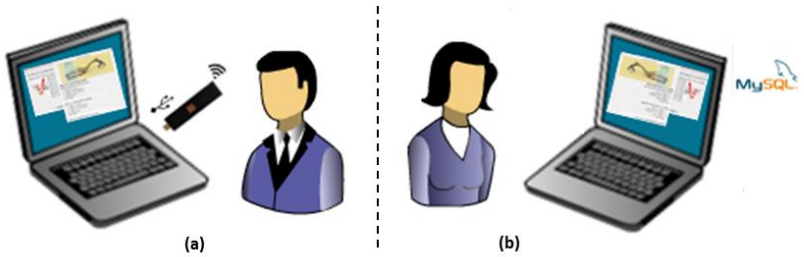

Figure IV – 7. Pre-Experimental Platform Architecture

The first group was given SPTs (1 per participant) containing the survey that they answer through this secure device (secure edition) (see Figure IV – 7a). All the answers remained in the SPT. However, authorized scores calculated using the weights of the given answers were disclosed. These scores did not reveal any sensitive information. The second group responded to the survey by connecting to a central server (vulnerable edition) (see Figure IV – 7b). All the answers have been stored in this server.

Both groups have been reassured that their sensitive information would securely be stored, and the same graphical interface has been used (see Figure IV – 12). The

survey administrators were responsible for system's initialization by providing the appropriate survey as input (see Figure IV – 8 and see Figure IV – 11).

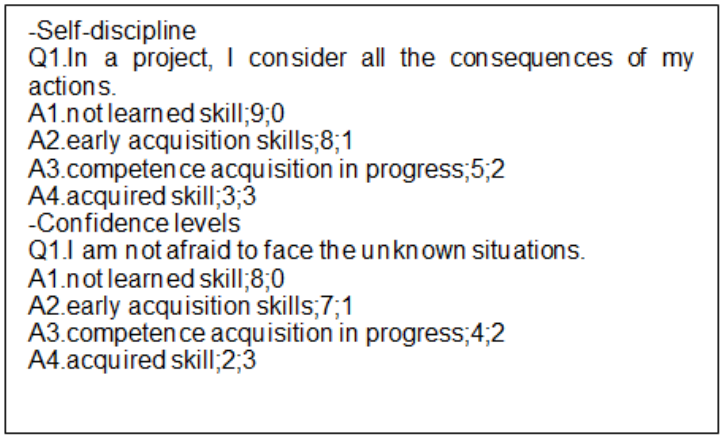

Figure IV – 8. Sample Questionnaire Survey

4.2.3.2 Hardware Used

In this pre-experiment, 70 SPT (1 SPT per student), connected to several terminals are used (see Figure IV – 7a). The drivers of the secure devices have also been installed in the terminals, while headsets were used so that the students could watch a short video presenting the concept of the secure devices that would be used in this pre-experiment. In this case, the students were given these devices, having the control of their sensitive data (data ownership). Furthermore, several stand-alone terminals, connected to the central server of the University of Paris-Saclay on which we had previously installed MySQL were used as well (see Figure IV – 7b).

4.2.3.3 Database Schema

For the questionnaire survey application, a common schema was developed in personal and central server. As we stated in Chapter 2, PlugDB was used for experimentation. Our DB schema was somewhat constrained since, for simplicity, we reused an existing DB schema. Figure IV - 9 depicts the database schema that our application used that contains the following tables: *Participant*, *Category*, *Question*, *Label*, and *QuestionInfo*.

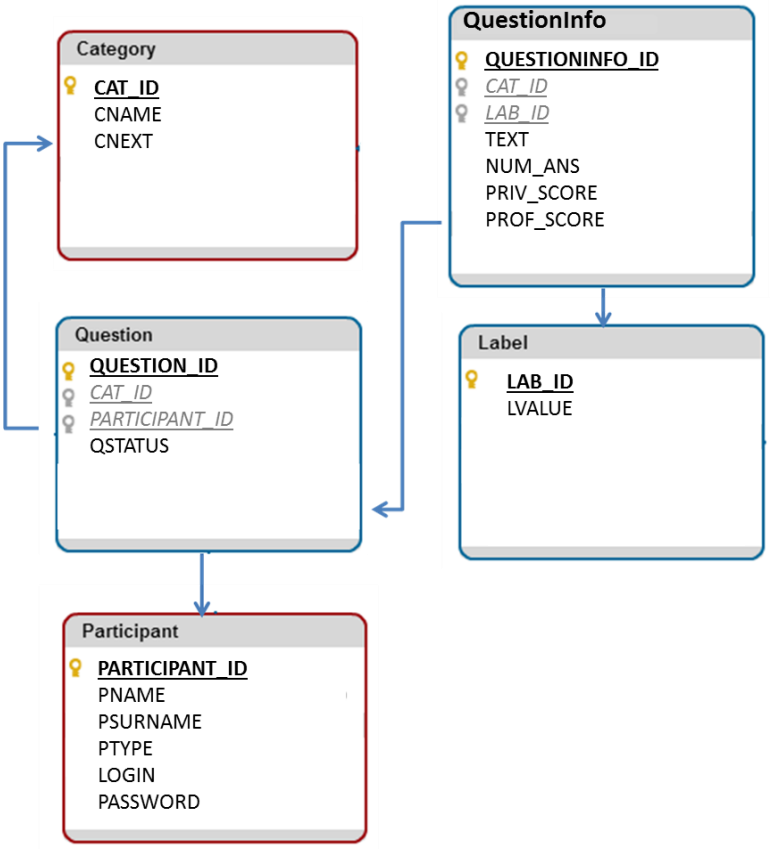

Figure IV – 9. Database Schema for Sensitive Questionnaires Survey

All tables include an id: *TABLE_ID*, which is the primary key. Participant table keeps the information that is related to the participant. PTYPE identifies the type of participant (1: student 0: administrator). *Category* table contains the questions' categories of the questionnaire. *CNAME* represents the name of questions' category while *CNEXT* links a category to the next one. Question table contains information related to the questions' status (answered/not answered). QSTATUS identifies if a question is answered or not (0: answered, -1: not-answered). Label table contains the labels of the questions/answers. LVALUE identifies if it is a question or an answer (0: question, 1 to 4: answer). QuestionInfo table contains the questionnaire. LABEL_ID contains either the question's or answer's id, TEXT keeps the free-text-response of the questionnaire and NUM_ANS keeps the number of answers for each question. PRIV_SCORE and the

PROF_SCORE keep the privacy and profile scores, accordingly, associated to an answer.

4.2.3.4 Creating a Survey

From the survey administrators' point of view, the system allowed managing the whole pre-experimental procedure. The procedure includes the building of the questionnaires, the initialization of the system and result monitoring. The results included a profile description that represents participants' skills and strengths. When the participants completed the process, the final score can be computed[4] based on profile scores of each question (the formulae being dependent of the questionnaire itself). The overall profile score occurred was the sum of the profile scores and provided a characterization of the user's profile based on that score (see Figure IV – 10).

The administrators could create the survey using a simple text editor. Figure IV – 8 shows a part of the questionnaire survey. The symbol '-' represents a new category, the letter: 'Q' represents a new question and the letter: 'A', a new answer. In order to perform the analysis, privacy and profile scores will be calculated. For this reason, after the desired answer, two numbers can be added followed by ';'. The first number represents the privacy and the second one the profile score. For instance, in Figure IV – 8, "A1.not learned skill;9;0"; 'A1' signifies that this is the first answer, "not learned skill" is the answer label, while the numbers 9 and 0 represent the privacy and profile score's values, accordingly. The administrators could transform this file into a csv file, the proper form of input file for system's initialization. When the experiment finished, the administrators could monitor the results (see Figure IV – 8 and Figure IV – 11). In the vulnerable version, a list of participants, along with their results was available to them for any processing.

[4] It is important to note that any complex (but local) computation could be run. The sum is merely used as a simple example. Global computations, such as SQL group by queries could be executed using the SQL/AA infrastructure (see To et al. [To16]).

-Adventurer
Enthusiasm is everywhere no matter the situation or experienced live! The student wants to be totally in control of his destiny. Success and wealth are both engines of his ambition.
-Realist
The project is carefully studied (cost estimates, constraints, timing and a very precise schedule).
-Prudent
The project is analyzed under the microscope with enthusiasm or pessimism depending on when the student will experiment before launching and taking action.
-Dreamer
Students live in an idealized world: the reality principle is not yet in the agenda of its concerns.

Figure IV – 10. Sample Profiles' Description

4.2.3.5 Answering the Survey

From the participants' point of view, the system provided information related to their potential future job, by answering a questionnaire survey. Figure IV – 12 exhibits the graphical interface from participants' side. The system allowed participants to create their accounts in order to login to the system and answer the questionnaire survey. The questionnaire was divided into categories that the participants were being asked to respond to. Once the participants have answered all the available questions (phase 1), they were able to see their results (phase 2), obtained after several calculations depending on the weights of the given answers.

4.2.3.6 Participants

One hundred forty students, who enrolled in an integration course, participated in the experiment. As we have already discussed, this pre-experimentation was conducted at the University of Paris-Sud XI in February 2016, in two phases. During the first phase the students were called to answer a questionnaire without knowing beforehand that this process was an experiment, while in the second phase the students were given their results, according to their answers. Note that each student concerning the Questionnaire Survey with SPTs, at the end of the phase 1 was given the SPT to take it

home (to ensure that their results are their properties) and was asked to bring it back for the next phase.

4.2.4 Full Experimentation protocol

In the future experimentation, the protocol could be enhanced with several informational shocks, which is a classical protocol in experimental economics. Examples of these shocks include error screens that will appear suddenly (i.e. virus effects, list with instructions for a limited time), actors impersonating survey organizers pretending something is wrong, etc. However, not all the participants will be experiencing the informational shocks. Each of the groups could be divided into two sub-groups. Table IV - 1 describes all the possible cases that would arise.

		Data Storage	
		Secure Portable Token	MySQL Server
Informational shock	Yes	Case1	Case2
	No	Case3	Case4

Table IV - 1. Experimentation protocol

Having added the shocks described, an exposure index could be calculated based on sensitive data that has been collected from all the participants (those who faced an informational shock and those who did not). The objective is to compare the index differences between the cases of participants having the SPT (cases 1 & 3) and those who are using the central server (cases 2 & 4), and observe whether owning a SPT, or being submitted to an informational shock, influences the index difference.

4.2.5 Prototype

Figure IV–11 shows the surveys' administrator platform in the vulnerable version.

Figure IV–12 shows the participation in the Questionnaire Survey.

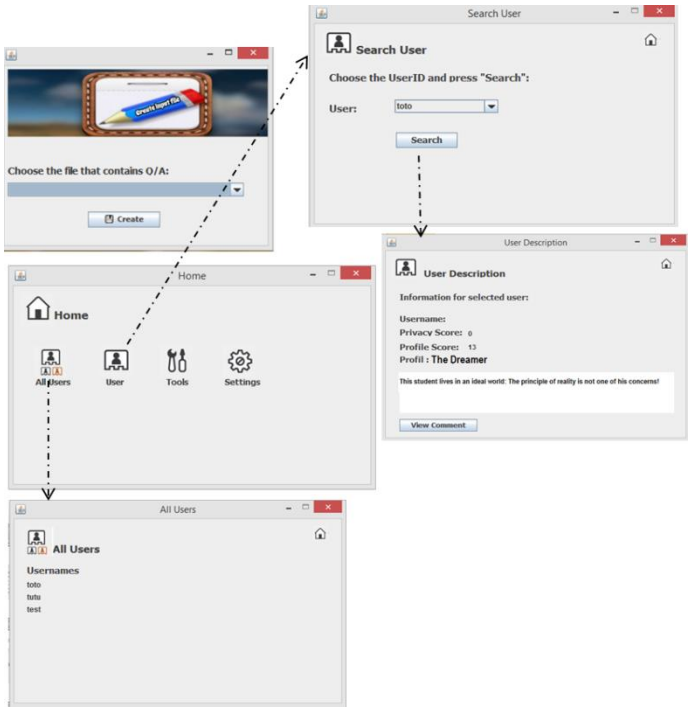

Figure IV – 11. GUI for Administrators in the Central server's version

Figure IV – 8 presents a sample of the questionnaire survey, created by the survey's administrators. This text file will be transformed into the appropriate form of our system's input file (see Figure IV – 11). The database initialization for both sides, PlugDB and MySQL server, was performed in the same manner. In the former case, the SPTs were plugged in a terminal one by one, and the SPTs' database was initialized (system module for SPT use chosen) (see Figure IV – 12). In the latter case, the same initialization process was followed, but it was made only once (system module for MySQL server use chosen). Once the participants have answered the survey, the profile and privacy scores along with the profiles' description was only visible by the survey's administrators (see Figure IV – 10 and Figure IV – 11), while their answers remained private.

Figure IV – 12. GUI for Participants

4.2.6 Findings and Limitations of the Existing Platform

Some of the limitations that we have faced during the experimentation process included:

- **Dedicated room**: Room reservation was required in order to conduct our experimentation. Assuming that the rate of use for the university facilities was very high, reservations are highly recommended and required in advance and therefore constrained the pre-experimentation. It is worth to mention that conducting the experimentation in a classic laboratory room was not possible for technical reasons such as USB ports that were not available, and thus the use of tokens was impossible. This issue is common in many universities and schools (e.g. at UVSQ, ENSIIE, INSA CVL).

- **Old-fashioned Equipment**: The computers that were used for the experimentation were too old to support virtual machines (i.e. VirtualBox), increasing the chance of repeated installation and software update. In addition, the old-fashioned hardware gave a negative image to the students.

- **Internet Access**: Internet access was necessary for conducting the pre-experimentation (transmission of the answers to the central server). In our case, wireless network of the University of Paris-Saclay was used, but it was a bit unstable.

In view of the above considerations, the success of experimentation, as well as its replication and improvement depend on the technical environment in which the experimentation will take place. A mobile laboratory could allow us to reduce the logistic costs and guarantee ergonomic questionnaire surveys. From an experimental point of view, a smaller screen size may increase the confidentiality of participants, since the answers are less visible to other participants that will be close[5]. In this direction, we can also consider a greater number of terminals (e.g. fifty tablets) to reduce the time of the experimentation, thus minimizing interactions between the participants.

4.2.7 Future Work on Sensitive Questionnaire Surveys

4.2.7.1 Principle of the Mobile Experimental Economy Laboratory

The principle of the proposed laboratory particularly addresses the limitations outlined above (see Section 2.6); seeking to meet the following objectives:

- **Mobility**: The laboratory must be transportable and autonomous. For instance, it should be able to fit in the trunk of a car, as well as it should be independent of the IT infrastructure of the laboratory (environment) where the experiment will take place.

[5] In experimental economics, panels are used in laboratory experiments to avoid group effects and ensure the participants for the confidentiality of their responses. However, in our case, we did not use such panels in order not to excite students' curiosity.

- **Simplified logistics**: It should support fast deployment; without complex and repeated manipulation.
- **Ergonomics**: It should correspond to current standards (i.e. applications on smartphones or tablets).
- **Internet Access /Internet Independence**: In the case where a connection would be available, then it should be easily sharable to the participants of the experimentation, as needed. In case there is no connection available, this laboratory should provide at least a local network where the participants can interact through a dedicated machine[6]. This local network will also allow us to do any appropriate configuration, as well as the deployment of the application on the tablets.

The realization of a secure and mobile laboratory of experimental economics requires the following:

- The acquisition of the mobile devices (Android tablets); allowing the collection of the personal information involved in the experimentation.
- The acquisition of a portable server, as well as a local network infrastructure (i.e. wireless routers).
- The implementation of an Android application; for collecting the data by questionnaires from a list of questions/ answers/ categories indicated directly in a text file by economists.
- The automation of the software deployment across the tablets and the personal servers, as well as the possession of the necessary tools for the maintenance of the mobile laboratory.

4.2.7.2 Technical description of the platform

Regarding the hardware, the proposed platform will consist of a set of Android tablets. A secure portable token will be connected to each of the aforementioned tablets through a

[6] In the experiment that was conducted in 2016, LAN was needed to store the experimental data to the dedicated server (version: central server).

USB cable; hosting personal data produced by the tablet's user (see Figure IV – 13d). Moreover, wireless communications will be established by wireless routers (see Figure IV – 13b) between the tablets and a server (powerful laptop). A secure portable token will be connected to the aforementioned server as well (see Figure IV – 13a), storing the statistical results. The calculation will be carried out to produce the statistical results of the experiment anonymously, without revealing the personal data entered by users and are stored in their secure portable tokens, respectively. The calculated statistical results will determine both the profile of each user and the overall level of sensitivity of the responses of a group of users. The results of each user will be accessible only to the user itself and will be calculated locally by his token while the overall statistical results - that will be calculated securely in distributed fashion by the tokens- will be accessible only to the economists that conduct the experiment. The tablets will be charged via multiport stations, allowing simultaneous charging (see Figure IV – 13c) and avoiding complex manipulations prior to experimentation. The platform does not include the equipment needed for loading the code into the secure portable tokens. The initial version of the experimentation included 70 tokens. These secure portable tokens are available to be reused.

Regarding the software, the platform will offer four main consoles that allow us to configure, deploy and conduct the experimentation, as well as to consult the statistical results. In particular, the first console will run on the server. The input of this console will be a simple text file that economists can write. This file will include questionnaires (sets of questions along with possible answers). The economists will specify the sequence of the questions, as well as assign two numerical values to each answer choice: the first value will be useful to calculate the profile of the user while the second value will evaluate the sensitivity of the response and thus, will help the economists to calculate the overall statistical results of the experiment for a group of participants. The aforementioned console will generate the Android application that will be used by individuals during the experimentation. The second console will deploy the Android application from the PC server to all the tablets, as well as it will initialize the database of each secure portable token. The third console will be used during the experimentation, such that the survey administrators can monitor its progress. The

fourth console will show to the economists the overall results of the experimentation, respecting the anonymity and privacy of the participants.

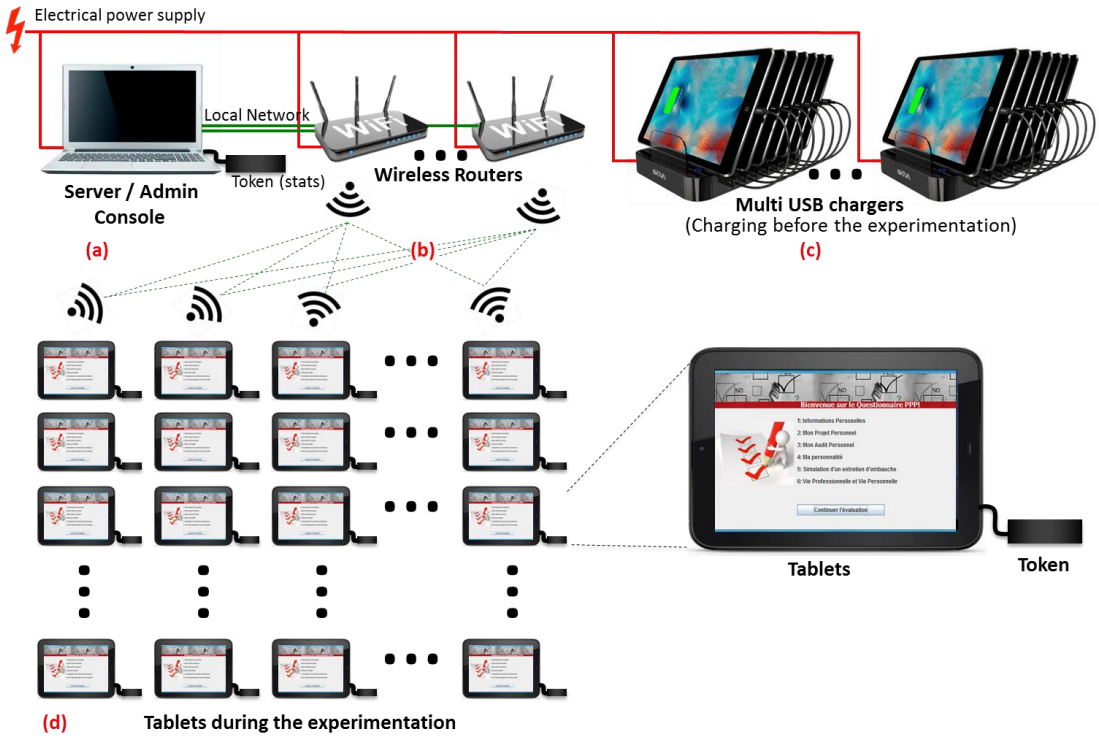

Figure IV – 13. Mobile-Laboratory Architecture

4.2.7.3 Valorization of the platform and laboratory

The setup of a mobile experimental laboratory will allow us to face the highlighted limitations in the pre-testing. More particularly, it will allow us to organize experimentations in various contexts and thus to test a larger number of more important research questions. For instance, in the short term, we can imagine experimentations on the impact of the environment, particularly the number of participants or the place that can be easily installed. These issues have not been addressed in the literature, because in the most cases the experimental economics laboratories are often fixed. This platform as well as the secure experimental economics laboratory can also be used for any experimentation that is dealing with sensitive information. We expect that this study will have positive impact both in terms of visibility by providing the scientific community with this platform (i.e. FabLab, University of Versailles) and usability in

various investigations of experimental economics related to personal data analysis. We will try to make this platform certified by the CNIL, the French data protection authority.

In the longer term, we can use this platform in order to test the usages in the mobile tablets that remain unknown in the academic world. Nowadays, only private parties have access to these data, partially [Lazer09]. Some experiments are currently in process and thus, expand the promising field of "Social Physics"[7] but not in cases where the personal data is collected anonymously. As far as the personal data is concerned, such data could thus allow us to highlight the impact of specific information (nudge) on user behavior.

4.2.8 Conclusion

In this section, we presented both secure and vulnerable approach of a system dedicated to Sensitive Questionnaire Surveys. We have designed pre-experimentation process with students in the context of proposing job-searching strategies, showing that the secure approach could potentially be fitted better to individuals. We have also described in details the pre-experimentation and experimentation protocols, the technical equipment and the system infrastructure that we have used to carry it out in collaboration with experimental economists, on groups of students. Finally, in future work, there are several interesting research opportunities for conducting the survey experimentation. Regarding Sensitive Questionnaire Surveys along with the Secure Portable Token (SPT) context, we plan to apply our experience from the pre-experimentation phase in order to organize better the experimentation. First, regarding the technical equipment, we will replace laptops by Android tablets. Second, we intend to enhance the questionnaire survey application ergonomic (specifically redesigned for android tablets). Third, we will provide students with better instructions, since our pre-experimentation showed that in some cases, students did not understand fully the concept of SPT. Given the above considerations, we hope to overcome the limitations of the existing infrastructure and to conduct the experimentation under better conditions.

[7]Projects OpenPDS, Funf (http://funf.org) and Reality Commons (MIT Media Lab, http://realitycommons.media.mit.edu/index.html).

Chapter V
Conclusion and Perspectives

In the era of information explosion, more and more people are connected online (e.g. OSN, emails, chat), any time from any device (e.g. smartphones, tablets, pc) such that the variety as well as the volume of digital records created, processed and analyzed is increasing dramatically. As the amount of devices along with the corresponding software exposed online increases, the amount of personal data (e.g. digital identity, communication logs, audio, photos, health data, financial data, insurance data) generated increases, as well. Thus, privacy and security of that data more than ever play an instrumental role, since both privacy and security are issues highly complex with multiple perspectives.

Individuals would intuitively like to have the control of who can access, use, aggregate, edit and share data about themselves. However, it is not always possible for them to be the exclusive owners of their personal information. For instance, regarding healthcare data, some medical providers are required to maintain in their systems certain records about patients. Another example could include OSN, where people are tagged, tracked and followed online. Few individuals are aware of the amount of data that they give away and how these data is going to be used and for what purposes.

Thus, there is a clear need of controlling the accesses and usages of this personal data. As described in previous sections, this thesis outlines a prototype implementation and an extended one based on well-known usage control model, as well as a first attempt to cover the domain of online sensitive questionnaire surveys by introducing a secure platform that manages sensitive data. The latter approach promoted in this thesis has been tested in the context of PAIP and Valdo projects over non-expert users.

This chapter concludes the thesis. We synthesize the work conducted, and close the manuscript by opening exciting research perspectives.

5.1 Synthesis

As stated in Chapter 1, we addressed in this thesis the problem of data sharing, as well as usage control issues. The goal of this thesis was multi-disciplinary in the context of the Digital Society Institute (ISN) of Paris-Saclay and its Privacy working group, which includes jurists, economists and computer scientists. Thus, one of the objectives was to study the issue of usage control through two axes: (a) computer science and (b) economic. More specifically, we combined computer science features such as algorithms, models and graphical interfaces with surveys with real users that fall into socio-economic domain in order to evaluate our concepts and systems, in terms of usability and adaptability.

The approaches proposed in this thesis are based on the conceptual usage control model; named $UCON_{ABC}$. First, we proposed a prototype implementation of the aforementioned model, applied to OSN scenario. Second, we presented an algebraic extension to $UCON_{ABC}$; called DatShA that allows a user to define privacy-preserving workflows on their data. Furthermore, we proposed an SPT based Data Sharing Platform used in the context of privacy laboratory classes, where the students were able to implement their own application on top of it, opening the way for the development of Privacy-by-design applications. We also referred to another experimental work that has been conducted in the context of the PAIP and Valdo projects on over one hundred forty non-expert users. In this work, we proposed a survey platform to manage sensitive data collected through online surveys, using this platform along with secure hardware (the SMIS secure Token and PlugDB server). Finally, we discussed the future work of this experimentation that includes a “mobile experimental lab”, in order to collect sensitive data for research purposes, while users’ privacy protection is guaranteed.

5.2 Perspectives

The work conducted in this thesis can be pursued in various directions. We identify below some challenging issues and outline possible lines of thought to tackle them.

- **DatShA in OSN**

  As we discussed in Chapter 3, using DatShA users can create ACPs in order to manage access control to their data. However, the power of this algebra can appear when users share ACPs between them. A future work could consider the case of an online marketplace of ACPs, as well as experimentations with real users to test its usability and adoptability. In this Market place, individuals, user associations or application developers, can define, use or rate operators or ACP according to their knowledge and experience.

- **Sensitive Questionnaire Surveys online (Website)**

  In the context of sensitive questionnaire surveys, a further study, therefore, could perform the experiment on a very large scale, including online questionnaires that contain questions related to security and privacy of personal information, as well as questions that inform the user about how individuals are using their personal information, the devices they use, as well as the applications to control their personal information.

  Conducting online surveys include many advantages. For instance, data collection is simpler and cheaper. The amount of participants may increase compared to the experimentation that we described in Chapter 4. The participants' answers will be automatically stored electronically and thus data analysis becomes easier.

  An initial basic website has already been implemented (see Appendix 3). Firstly, this website welcomes the participants of the survey by providing them the purpose of the web site and a brief definition of Personal information. In addition, the participants are able to read some statistics related to security of personal data (e.g. sharing/storing/editing data, usage of personal data from third parties). The main idea of this page is to deliver a message to the participants such that at the end of their participation, they will be able to be aware of what is happening

with their personal data and what are the existing techniques to keep their data safe.

A first set of questions related to how they create, use, modify and store their personal data, as well as the type of devices they use and the applications installed on them is available to the participants. Having answered these first questions, then a first profile description that corresponds to the questions answered is displayed. Then, a variety of informational shocks using examples from everyday life are shown. The participation of the questionnaire will end-up by a second set of questions slightly different, compared to first one, in order to compare the differences between their initial and second set of answers. The participants will be given a final profile description[1] at the end of the questionnaire, and some tips to control better his data.

This work has already started and may continue in collaboration with Amelie Coulbaut, a sociologist in David Lab. She may help performing a social study and selecting carefully the questions that should be included in our online survey, in order to achieve good response rates and fair questionnaire (without influencing the user).

- **Sensitive Questionnaire Surveys (Mobile Laboratory)**

  As we discussed in Chapter 4, an ongoing work considers the case of a secure and mobile laboratory that includes mobile devices (Android tablets), portable servers (Secure Portable Tokens), an Android application that collects the data from the questionnaires forms as well as tools that provide easy application deployment across the tablets and maintenance of the mobile laboratory. In the light of the aforementioned considerations, this mobile laboratory will allow us to set up experimentations in various contexts, including larger amount of research questions (e.g. environment, energy, smart cities) as well as larger amount of participants. We have already bought seventy android tablets and a master student is working on the adequate software implementation (as described in Chapter 4), able to be executed in these devices, in order to carry out the concept of Mobile Laboratory.

[1]Both first and second set's answers count in order to define the final profile description.

Bibliography

[Ahern07] Ahern, Shane, Dean Eckles, Nathaniel S. Good, Simon King, Mor Naaman, and Rahul Nair. "Over-exposed?" Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07 (2007): 357-366.

[Allard10] Allard, Tristan, Shaoyi Yin, Nicolas Anciaux, Luc Bouganim, Yanli Guo, Lionel Le Folgoc, Benjamin Nguyen, Philippe Pucheral, Indrajit Ray, and Indrakshi Ray. "Secure personal data servers." Proceedings of the VLDB Endow 3.1-2 (2010): 25-35.

[Allard11] Allard, Tristan, Benjamin Nguyen, and Philippe Pucheral. "Safe realization of the Generalization privacy mechanism." Proceedings of the Ninth Annual International Conference on Privacy, Security and Trust (2011).

[Allard14] Allard, Tristan, Benjamin Nguyen, and Philippe Pucheral. "MET_AP: revisiting Privacy-Preserving Data Publishing using secure devices." Distributed Parallel Databases 32.2 (2013): 191-244.

[Andrews03] Andrews, Dorine, Blair Nonnecke, and Jennifer Preece. "Electronic Survey Methodology: A Case Study in Reaching Hard-to-Involve Internet Users." International Journal of Human-Computer Interaction 16.2 (2003): 185-210.

[Anciaux08] Anciaux, Nicolas, Morgane Berthelot, Laurent Braconnier, Luc Bouganim, Martine De la Blache, Georges Gardarin, Philippe Kesmarszky, Sophie Lartigue, Jean-François Navarre, Philippe Pucheral, Jean-Jacques Vandewalle, and Karine Zeitouni. "A Tamper-Resistant and Portable Healthcare Folder." International Journal of Telemedicine and Applications 2008 (2008): 1-9.

[Anciaux10] Anciaux, Nicolas, Luc Bouganim, Yanli Guo, Philippe Pucheral, Jean-Jacques Vandewalle, and Shaoyi Yin. "Pluggable personal data servers." Proceedings of the 2010 international conference on Management of data - SIGMOD '10 (2010).

[Anciaux13] Anciaux, Nicolas, Luc Bouganim, Benjamin Nguyen, Iulian S.U Popa. "Trusted Cells: A Sea Change for Personal Data Services." 6th Biennial Conference on Innovative Data Systems Research (CIDR ’13) Asilomar, California, USA, 6-9 January, 2013.

[Anciaux15a] Anciaux, Nicolas, Luc Bouganim, Philippe Pucheral. "Plans Hardware du Token PlugDB." France, N° de brevet: Enregistrement APP no.IDDN.FR.001.090013.000.S.P.2015.000.20600. 2015.

[Anciaux15b] Anciaux, Nicolas, Luc Bouganim, Philippe Pucheral, Shaoyi Yin, Quentin Lefebvre, et al. "Logiciel PlugDB-engine version 4." France, N° de brevet: Enregistrement APP no.IDDN.FR.001.280004.000.S.C.2008.0000.10000. 2015.

[Arrison04] Arrison, Sonia. "Is Friendster the new TIA? TechCentralStation." 7 January, 2004.

[Balfanz00] Balfanz, Dirk, Drew Dean, and Mike Spreitzer. "A security infrastructure for distributed Java applications." Proceeding of the 2000 IEEE Symposium on Security and Privacy. (2000): 15-26.

[Beresnevichiene03] Beresnevichiene, Yolanta. "A role and context based security model" (2003).

[Bier13] Bier, Christoph. "How Usage Control and Provenance Tracking Get Together - A Data Protection Perspective." Proceeding of the 2013 IEEE Security and Privacy Workshops (2013): 13–17.

[Black04] Black,John "The perils and promise of online schmoozing." Business Week Online, February, (2004).

[Blaze96] Blaze, Matt, Joan Feigenbaum, and Jack Lacy, "Decentralized Trust Management." Proceeding of the 1996 IEEE Symposium on Security and Privacy. IEEE Computer Society, Washington, DC, USA (1996): 164-173.

[Bonneau09] Bonneau, Joseph, Jonathan Anderson, and Luke Church. "Privacy suites: Shared privacy for social networks." Proceedings of the 2009 Symposium On Usable Privacy and Security, Mountain View, CA, USA, 15 July, 2009.

[Bonneau09] Bonneau, Joseph, Soren Preibusch. "The privacy jungle: On the market for data protection in social networks." Proceedings of the 8th workshop on the Economics of Information Security, June, 2009.

[Bouganim16] Bouganim, Luc, Athanasia Katsouraki, Benjamin Nguyen. "DatShA : A Data Sharing Algebra for access control plans." Proceedings of the EDBT 2016, (2016): 710-711.

[Boyd03] Boyd, Danah. "Reflections on Friendster, Trust and Intimacy." Proceedings of the Ubiquitous Computing Workshop 2003, Seattle, Washington, USA, 2003.

[Boyd04] Boyd, Danah. "Friendster and publicly articulated social networking." Extended abstracts of the 2004 conference on Human factors and computing systems - CHI '04 (2004).

[Boyd07] Boyd, Danah, Nicole Ellison. "Social Network Sites: Definition, History, and Scholarship." Journal of Computer-Mediated Communication 13.1 (2007): 210-230.

[Calo14] Calo, Ryan. “Code, Nudge, or Notice.” Iowa Law Review 99 (2014): 773, 775–90.

[Carminati09] Carminati, Barbara, Elena Ferrari, and Andrea Perego. "Enforcing access control in Web-based social networks." ACM Transactions on Information and System Security 13.1 (2009): 1-38.

[Carniani16] Carniani, Enrico, Davide D’Arenzo, Aliaksandr Lazouski, Fabio Martinelli, and Paolo Mori. "Usage Control on Cloud systems." Future Generation Computer Systems 63 (2016): 37-55.

[Chadwick10] Chadwick, David, Kaniz Fatema. “Distributed Privacy Policy Enforcement by using Sticky Policies” University of Kent, 2010.

[Danezis09] Danezis, George. "Inferring privacy policies for social networking services." Proceedings of the 2nd ACM workshop on Security and artificial intelligence, Chicago, Illinois, USA, 2009.

[DBS] Databases, available Online: http://www.cs.nott.ac.uk/~psznza/G51DBS/dbs19.pdf.

[Dayan07] Dayan, Yehuda. “Responding to sensitive questions in surveys: A comparison of results from Online panels, face to face, and self-completion interviews.” World Association for Public Opinion Research, Berlin, 2007.

[Denning76] Denning, Dorothy E. "A lattice model of secure information flow." Communications of the ACM19.5 (1976): 236-243.

[Doyle14] Doyle, James K. "Face-to-Face Surveys." Wiley StatsRef: Statistics Reference Online, 2014.

[Eisenberg13] Eisenberg, Andrew. "XQuery 3.0 is nearing completion." ACM SIGMOD Record 42.3 (2013): 34-41.

[Ellison99] Ellison, Carl, Bill Frantz, Butler Lampson, Ron Rivest, Brian Thomas, and Tatu Ylonen. "SPKI Certificate Theory." 1999.

[Ferraiolo03] Ferraiolo, David, Richard Kuhn, Ramaswamy Chandramouli. “Role-Based Access Control.” Artech House, Norwood, MA, USA, 2003.

[Goffman59] Goffman, Erving. “The Presentation of Self in Everyday Life.” New York: Doubleday, 1959.

[Gooch03] Gooch, Richard. "Requirements for DRM Systems." Lecture Notes in Computer Science (2003): 16-25.

[Grompanopoulos13] Grompanopoulos, Christos, Antonios Gouglidis, and Ioannis Mavridis. "A Use-Based Approach for Enhancing UCON." Security and Trust Management (2013): 81-96.

[Gross05] Gross, Ralph, Alessandro Acquisti, and H. Heinz. "Information revelation and privacy in online social networks." Proceedings of the 2005 ACM workshop on Privacy in the electronic society, 2005.

[Guth03] Guth, Susanne. "A Sample DRM System." Lecture Notes in Computer Science (2003): 150-161.

[Halderman04] Halderman, J. A., Brent Waters, and Edward W. Felten. "Privacy management for portable recording devices." Proceedings of the 2004 ACM workshop on Privacy in the electronic society, 2004.

[Hawkey06] Hawkey, Kirstie, and Kori M. Inkpen. "Keeping up appearances." Proceedings of the SIGCHI conference on Human Factors in computing systems, 2006.

[Herzberg00] Herzberg, Amir, Yosi Mass, Joris Mihaeli, Dalit Naor, and Yiftach Ravid. "Access control meets public key infrastructure, or: assigning roles to strangers." Proceedings of the 2000 IEEE Symposium on Security and Privacy (2000): 2-14.

[House06] House, Van, Nancy, Marc Davis, Morgan Ames, Megan Finn, Vijay Viswanathan. "The uses of personal networked digital imaging." CHI '06 extended abstracts on Human factors in computing systems (2005): 1853-1856.

[Hu11] Hu, Vincent, Richard Kuhn, Tao Xie, and Jee Hyun Hwang. “Model checking for verication of mandatory access control models and properties.” International Journal of Software Engineering and Knowledge Engineering, 21.1 (2011): 103-127.

[Hu13] Hu, Hongxin, Gail-Joon Ahn, and Jan Jorgensen. "Multiparty Access Control for Online Social Networks: Model and Mechanisms." IEEE Trans. Knowl. Data Eng 25.7 (2013): 1614-1627.

[Ito05] Ito, Mizuko. “Personal, Portable, Pedestrian: Mobile Phones in Japanese Life.” MIT Press, 2005.

[Katsouraki15a] Katsouraki, Athanasia, Luc Bouganim, Benjamin Nguyen, Paul Tran-Van, Secure Portable Tokens for Sensitive Questionnaires Surveys, 6e Atelier sur la Protection de la Vie Privée. Demo paper, Mosnes, France, 2015.

[Katsouraki15b] Katsouraki, Athanasia, Luc Bouganim, Benjamin Nguyen, Paul Tran-Van, Secure Portable Tokens for Sensitive Questionnaires Surveys, 31èmes journées Bases de Données Avancées. Demo paper, 2015, Île de Porquerolles, France, 2015.

[Kelbert12] Kelbert, Florian, and Alexander Pretschner. "Towards a policy enforcement infrastructure for distributed usage control." Proceedings of the 17th ACM symposium on Access Control Models and Technologies, 2012.

[Kindberg05] Kindberg, Tim, Mirjana Spasojevic, Rowanne Fleck, Abigail Sellen. "The Ubiquitous Camera: An In-Depth Study of Camera Phone Use." IEEE Pervasive Comput 4.2 (2005): 42-50.

[Krishnamurthy09] Krishnamurthy, Balachander, and Craig E. Wills. "On the leakage of personally identifiable information via online social networks." Proceedings of the 2nd ACM workshop on Online social networks, 2009.

[Lampson74] Lampson, Butler, Weina Ge, Divya Muthukumaran. “Protection.” SIGOPS Operating Systems Review, Volume 8, Issue 1 (1974): 18-24.

[Lazer09] Lazer, David, Alex Pentland, Lada Adamic, Sinan Aral, Albert-László Barabási, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, Tony Jebara, Gary King, Michael Macy, Deb Roy, Marshall Van Alstyne. “Computational Social Science”, Science (2009): 721-723.

[Leonard04] Leonard, Andrew. “You are who you know.” available Online: www.salon.com. 2004.

[Li11] Li, Ninghui. “Discretionary access control.” Encyclopedia of Cryptography and Security, Springer, US, (2011): 353-356.

[Liu03] Liu, Qiong, Reihaneh Safavi-Naini, Nicholas Paul Sheppard. “Digital rights management for content distribution.” Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003. Darlinghurst, Australia, (2003): 49-58.

[Manzano13] González-Manzano, Lorena, Ana González-Tablas, José De Fuentes, “Security and Privacy Preserving in Social Networks.” Springer, In press 2013, Chapter User-Managed Access Control in Web Based Social Networks, 2013.

[Manzano14] González-Manzano, Lorena, Ana González-Tablas, José De Fuentes, and Arturo Ribagorda. "SoNeUCON$_{ABC}$, an expressive usage control model for Web-Based Social Networks." Computers & Security 43 (2014): 159-187.

[Mao11] Mao, Ziqing, Ninghui Li, Hong Chen, and Xuxian Jiang. "Combining Discretionary Policy with Mandatory Information Flow in Operating Systems." ACM Transactions on Information and System Security 14.3 (2011): 1-27.

[Matsuo04] Matsuo, Hisako, Kevin McIntyre, Terry Tomazic, Barry Katz. “The Online Survey: Its Contributions and Potential Problems.” JSM 2004, Toronto, 2004.

[Matthews03] Matthews, Mark, Jim Cole, and Joe Gradecki. Mysql and Java Developer's Guide. Indianapolis: Wiley, 2003.

[Mazurek14a] Mazurek, Michelle. "A Tag-Based, Logical Access-Control Framework for Personal File Sharing." (2014). Dissertations.

[Mazurek14b] Mazurek, Michelle, Yuan Liang, William Melicher, Manya Sleeper, Lujo Bauer, Gregory Ganger, Nitin Gupta, Michael K. Reiter. “Toward strong, usable access control for shared distributed data.” Proceedings of the 12th USENIX Conference on File and Storage Technologies, 2014.

[McManus13] McManus, Sean, and Mike Cook. Raspberry Pi for Dummies. Hoboken: John Wiley & Sons, 2013.

[Milgram77] Milgram, Stanley. “The familiar stranger: An aspect of urban anonymity.” Individual in a Social World: Essays and Experiments, 1977.

[Mont03] Mont, Casassa, Siani Pearson, and Pete Bramhall. "Towards accountable management of identity and privacy: sticky policies and enforceable tracing services." Proceedings of the 14th International Workshop on Database and Expert Systems Applications, 2003. available Online: hhtp://www.hpl.hp.com/techreports/2003/HPL-2003-49.pdf.

[Monteleone15] Monteleone, Shara, René van Bavel, Nuria Rodríguez-Priego, Gabriele Esposito. “Nudges to Privacy Behaviour: Exploring an Alternative Approach to Privacy Notices.” JRC Science and Policy Report, 2015.

[Newitz03] Newitz, Annalee. “Defenses lacking at social network sites.” Security Focus, 2003.

[NCSCUS87] National Computer Security Center (U.S.). "A Guide to Understanding Discretionary Access Control in Trusted Systems." The ‘Orange Book’ Series (1987): 659-693.

[Osborn97] Osborn, Sylvia. "Mandatory access control and role-based access control revisited."Proceedings of the second ACM workshop on Role-based access control - RBAC '97 (1997): 31-40.

[Ong00] Ong, Anthony, and David Weiss. "The Impact of Anonymity on Responses to Sensitive Questions1." Journal of Applied Social Psychology 30.8 (2000): 1691-1708.

[Park02a] Park, Jaehong, and Ravi Sandhu. "Originator control in usage control." Proceedings of the Third International Workshop on Policies for Distributed Systems and Networks, 2002.

[Park02b] Park, Jaehong, Ravi Sandhu. “Towards usage control models: beyond traditional access control.” Proceedings of the seventh ACM symposium on Access control models and technologies (2002): 57-64.

[Park04] Park, Jaehong, Ravi Sandhu. 2004. The $UCON_{ABC}$ usage control model. ACM Trans. Inf. Syst. Secur. 7.1 (2004): 128-174.

[Paskin03] Paskin, Norman. “Identication and metadata.” Digital Rights Management (2003):26-61.

[PersonalData11] Personal Data: The Emergence of a New Asset Class , An Initiative of the World Economic Forum. 2011, available Online: http://www3.weforum.org/docs/WEF_ITTC_PersonalDataNewAsset_Report_2011.pdf.

[Piez99] Piez, Wendell. "XML: The Annotated Specification by Bob DuCharme." Markup Languages: Theory and Practice 1.3 (1999): 115-115.

[Qinlong14] Qinlong, Huang, Ma Zhaofeng, Yang Yixian, Niu Xinxin, and Fu Jingyi. "Improving security and efciency for encrypted data sharing in online social networks." China Communications 11.3 (2014): 104-117.

[Rump03] Rump, Niels. "Digital Rights Management: Technological Aspects." Lecture Notes in Computer Science (2003): 3-15.

[Saaya07] Saaya, Zurina, Anusuriya Devaraju, Nuridawati Mustafa, Chew Choon Leong. “The implementation of Questionnaires Design Principles via online questionnaire builder.” 2007.

[Samarati01] Samarati, Pierangela, Sabrina De Capitani di Vimercati. "Access control: Policies, models, and mechanisms." Tutorial Lectures, FOSAD '00 (2001): 137-196.

[Sandhu96a] Sandhu, Ravi. "Role hierarchies and constraints for lattice-based access controls." Proceedings of the Fourth European Symposium on Research in Computer Security (1996): 65-79.

[Sandhu96b] Sandhu, Ravi, Edward Coyne, Hal Feinstein, Charles Youman. “Role-based access control models.” 29.2 (1996): 38-47.

[Schaniel14] Schaniel, Ronnie. “Design and Implementation of an Online Questionnaire Tool.” ETH Master Thesis, 2014.

[Sege05] Sege, I. “Where everybody knows your name.” The Boston Globe. 2005, available Online: http://www.boston.com/.

[Shafi11] Shafi, Abdulhamid, Ahmad Sulaiman, Waziri Victor, Jibril Fatima. “Privacy and National Security Issues in Social Networks: The Challenges.” International Journal of the Computer, the Internet and Management 19.3 (2011): 14 -20.

[Shan12] Shan, Zhiyong, Xin Wang, and Tzi-cker Chiueh. "Enforcing Mandatory Access Control in Commodity OS to Disable Malware." IEEE Transactions on Dependable and Secure Computing 9.4 (2012): 541-555.

[Spiekermann01] Spiekermann, Sarah, Jens Großklags, and Bettina Berendt. "Stated Privacy Preferences versus Actual Behaviour in EC Environments: a Reality Check." e-Finance (2001): 129-147.

[SQLManual] SQL Manual, available Online: http://sqlpro.developpez.com/cours/sqlaz/ddl/?page=partie1#L0.

[SQLiteManual] SQLite Manual, available Online: https://www.sqlite.org/foreignkeys.html.

[SQLW3C] SQL W3C, available Online: http://www.w3schools.com/sql/.

[Tang08] Tang, Qiang. “On Using Encryption Techniques to Enhance Sticky Policies Enforcement.” Centre for Telematics and Information Technology, University of Twente, 2008.

[Thaler08] Thaler, Richard, Cass Sustein "Nudge: Improving Decisions About Health, Wealth, and Happiness." Yale University Press, 2008.

[To14] To, Cuong-Quoc, Benjamin Nguyen, Philippe Pucheral. “Privacy-Preserving Query Execution using a Decentralized Architecture and Tamper Resistant Hardware.” 17th International Conference on Extending Database Technology, 2014.

[To16] To, Cuong-Quoc, Benjamin Nguyen, Philippe Pucheral. “Private and Scalable Execution of SQL Aggregates on a Secure Decentralized Architecture”, Transactions on Database Systems 41.3 (2016): 46 pages.

[Tourangeau07] Tourangeau, Roger, Ting Yan. "Sensitive questions in surveys." Psychological Bulletin 133.5 (2007): 859-883.

[VHEA96] Varian, Hal. "Economic Aspects of Personal Privacy." Internet Policy and Economics (2009): 101-109.

[Wang11] Wang, Yang, Gregory Norcie, Saranga Komanduri, Alessandro Acquisti, Pedro G. Leon, and Lorrie F. Cranor. "I regretted the minute I pressed share." Proceedings of the Seventh Symposium on Usable Privacy and Security, 2011.

[Wang13] Wang, Yang, Pedro G. Leon, Kevin Scott, Xiaoxuan Chen, Alessandro Acquisti, and Lorrie F. Cranor. "Privacy nudges for social media." Proceedings of the 22nd International Conference on World Wide, 2013.

[Wang14] Wang, Yang, Pedro G. Leon, Alessandro Acquisti, Lorrie F. Cranor, Alain Forget, and Norman Sadeh. "A field trial of privacy nudges for facebook." Proceedings of the 32nd annual ACM conference on Human factors in computing systems, 2014.

[Watts03] Watts, Duncan. "Six degrees: the science of a connected age." Choice Reviews Online 40.11 (2003): 40-6452-40-6452.

[Weizenbaum66] Weizenbaum, Joseph. "ELIZA --- a computer program for the study of natural language communication between man and machine." Communications of the ACM 9.1 (1966): 36-45.

[Wobber93] Wobber, Edward, Martín Abadi, Michael Burrows, and Butler Lampson. "Authentication in the Taos operating system." ACM Transactions on Computer Systems 12.1 (1994): 3-32.

[XACMLIntro] Brief Introduction to XACML, available Online: http://www.oasis-open.org/committees/download.php/2713/Brief_Introduction_to_XACML.html.

[Zhao07] Zhao, Yong, Mihael Hategan, Ben Clifford, Ian Foster, Gregor Von Laszewski, Veronika Nefedova, Ioan Raicu, Tiberiu Stef-Praun, and Michael Wilde. "Swift: Fast, Reliable, Loosely Coupled Parallel Computation." 2007 IEEE Congress on Services (Services 2007) (2007): 199–206.

Appendix

1. QUESTIONNAIRES PPP

Le cours de PPP - Projet Professionnel Personnalisé - vous propose de répondre à différents questionnaires en vue de vous aider à formuler votre projet personnel et professionnel et vous proposer une réponse personnalisée. Vous êtes invités à répondre aux questions qui suivent. D'ordinaire, ces questions sont posées en face-à-face pour vous aider à élaborer vous-même votre projet professionnel et à conclure sur votre autonomie face aux grandes comme aux petites décisions de votre vie. Les questions posées vous permettent d'obtenir un score indiquant votre niveau d'ouverture pour vous auto-évaluer. Cela vous prendra entre 15 et 25 minutes.

Les informations recueillies peuvent faire l'objet d'un traitement statistique destiné à comprendre et évaluer votre objectif professionnel. Conformément à la loi « informatique et libertés » du 6 janvier 1978 modifiée en 2004, vous bénéficiez d'un droit d'accès et de rectification des informations vous concernant.

CAS 1: Vos données sont stockées sur un serveur de l'Université Paris-Sud et vous pouvez y accéder depuis cet ordinateur.

CAS 2: Vos données sont stockées sur une clé USB sécurisée, c'est-à-dire qu'en cas de perte ou de vol vos données personnelles sont inaccessibles par un tiers. Puisqu'il s'agit d'un outil techniquement novateur, nous vous proposons de visionner la vidéo suivante pour en comprendre le principe, ainsi que le fonctionnement. A la fin de cette séance, vous gardez la clé. Merci de la ramener pour les séances suivantes. Ce dispositif est en cours d'évaluation et est donc proposé uniquement dans le cadre de certains groupes de PPP.

1. INFORMATIONS PERSONNELLES

Nom :
Prénom :
Date de naissance (jj/mm/aaaa) :
Sexe :

- Homme
- Femme
- Je ne souhaite pas répondre

2. MON PROJET PERSONNEL

1. En quelle année avez-vous passé votre baccalauréat ?

- En 2012
- En 2013
- En 2014
- Autre : ...
- Je ne souhaite pas répondre.

… dans quel pays ?

- En France
- Autre : …
- Je ne souhaite pas répondre.

… dans quelle filière ?

- L
- S
- E
- Autre : …
- Je ne souhaite pas répondre.

2. Dans quelle formation étiez-vous inscrit l'année dernière ?

- L1 Economie-Gestion
- L2 Economie-Gestion
- Prépa économique
- Autre : …
- Je ne souhaite pas répondre.

3. Dans quelle formation souhaitez-vous vous inscrire l'année prochaine ?

- L3 Economie Appliquée
- L3 Gestion
- L3 Comptabilité
- L3 en alternance
- École de commerce
- J'arrêterai mes études
- Autre : …
- Je ne souhaite pas répondre.

4. Sur l'escalier suivant, la marche 1 correspond à la place la moins élevée dans la société et la marche 10 à la place la plus élevée. Sur quelle marche de cet escalier vous placeriez-vous ?

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- Je ne souhaite pas répondre.

5. Avez-vous changé d'orientation ?

- Oui, je m'étais trompé(e) de voie.
- Oui, j'ai échoué aux examens.
- Oui, je me suis passionné(e) tout à coup pour l'économie.
- Oui, j'ai déménagé.
- Oui, mais je ne sais pas pourquoi.
- Non
- Je ne souhaite pas répondre.

6. Pourquoi avez-vous choisi votre formation actuelle ? (choisissez la raison la plus importante parmi les réponses)

- Cette formation permet de trouver du travail.
- Cette formation donne des bases solides en économie.
- Cette formation permet d'avoir des outils pour changer le monde.
- Cette formation permet de réussir les concours pour les écoles de commerce.
- Cette formation est bien cotée .
- Les enseignants de cette formation sont exceptionnels.
- Je n'ai pas choisi : c'est la seule université qui m'a accepté.
- Je ne sais pas.
- Autre : …
- Je ne souhaite pas répondre.

7. Avez-vous identifié la sphère professionnelle dans laquelle vous désirez vous engager ?

- Oui, depuis longtemps.
- Oui, depuis mon entrée en Licence.
- Non, mais j'espère que le module PPP m'y aidera.
- Non, mais j'ai tout mon temps.
- Non, mais je suis ouvert à tout.
- Autre : …
- Je ne souhaite pas répondre.

8. Avez-vous identifié précisément le métier que vous souhaitez exercer ?

- Oui, depuis longtemps.
- Oui, depuis mon entrée en Licence.
- Non, mais j'espère que le module PPP m'y aidera.
- Non, mais j'ai tout mon temps.
- Non, mais je suis ouvert à tous les métiers.
- Autre : …
- Je ne souhaite pas répondre.

9. Pouvez-vous identifier des compétences opérationnelles ou techniques acquises au cours de vos études ?

- Oui, mais je n'ai acquis aucune compétence technique.
- Oui, mais la faculté ne nous permet pas d'en acquérir.
- Non, pas du tout.
- Non, pas vraiment.
- Non, pas toutes.
- Je ne sais rien faire.
- Autre : …
- Je ne souhaite pas répondre.

3. MON AUDIT PERSONNEL

Avez-vous l'esprit d'initiative ? L'audit personnel permet à chaque étudiant d'identifier ses points forts et ses points faibles. En prenant exemple dans votre vie personnelle ou dans votre vie universitaire, vous pouvez répondre à des questions comme : Qui a choisi votre voie ? Êtes-vous sûr que votre projet professionnel vous ressemble ? Quelle est votre culture professionnelle familiale ? Savez-vous être autonome dans vos décisions ? Êtes-vous force de proposition ? Êtes-vous leader ou suiveur ?

Le questionnaire suivant évoque huit situations liées à une compétence à identifier. Vous devez noter chacune de ces propositions 1, 2, 3, 4 et 5 selon l'échelle suivante :
1 : Pas du tout d'accord
2 : Plutôt pas d'accord
3 : Plutôt d'accord
4 : D'accord
5 : Tout à fait d'accord
NP : Ne souhaite pas répondre

1. Au moment de lancer un projet (projet professionnel, soirée entre amis, création d'activité, voyage, etc.)

	1	2	3	4	5	NP
Je sais construire n'importe quel projet.	○	○	○	○	○	○
J'ai la vision globale de chaque projet.	○	○	○	○	○	○
J'ai la capacité de construire seul un projet.	○	○	○	○	○	○
Je sais mener une étude de marché avant de me lancer dans un projet.	○	○	○	○	○	○
J'ai la capacité de faire évoluer un projet.	○	○	○	○	○	○

2. Du côté de vos finances et de votre gestion de l'argent

	1	2	3	4	5	NP
Je sais comment gagner de l'argent.	○	○	○	○	○	○
J'ai de l'aisance face à un banquier pour négocier un prêt.	○	○	○	○	○	○
Je gère mon budget, si modeste soit-il.	○	○	○	○	○	○
Quand j'ai des problèmes d'argent je me Débrouille pour trouver un petit boulot.	○	○	○	○	○	○
Avant même de gagner de l'argent, je sais déjà comment le dépenser.	○	○	○	○	○	○

3. En société ou dans un travail en équipe

	1	2	3	4	5	NP
J'ai le sens de l'humour.	○	○	○	○	○	○
Je suis parfaitement à l'aise avec les gens qui ont mieux réussi que moi.	○	○	○	○	○	○
J'ai une bonne mémoire des noms.	○	○	○	○	○	○
Je sais motiver les autres.	○	○	○	○	○	○
Je sais écouter.	○	○	○	○	○	○

4. Votre confiance en vous

	1	2	3	4	5	NP
Je sais me contrôler.	○	○	○	○	○	○
Je suis ambitieux.	○	○	○	○	○	○
J'aime les défis, les challenges.	○	○	○	○	○	○
Je ne crains pas d'affronter les situations inconnues.	○	○	○	○	○	○
J'ai de la fierté.	○	○	○	○	○	○

5. Votre goût du risque

	1	2	3	4	5	6
J'ai plus la mentalité d'un joueur de hasard que d'un joueur de réflexion.	○	○	○	○	○	○
Je suis opportuniste dans tous les sens du terme.	○	○	○	○	○	○
Je n'ai pas peur de l'échec.	○	○	○	○	○	○
Je suis prêt à soulever des montagnes lorsque je crois à mes idées.	○	○	○	○	○	○
J'aime découvrir de nouveaux domaines, aborder de nouveaux sujets.	○	○	○	○	○	○

6. Questions de pugnacité

	1	2	3	4	5	NP
Je suis têtu.	○	○	○	○	○	○
Pour atteindre un objectif, je suis capable de me priver, de souffrir.	○	○	○	○	○	○
J'ai plus une mentalité de coureur de fond que de sprinter.	○	○	○	○	○	○
Je n'aime pas rester sur un échec, quitte à recommencer plusieurs fois.	○	○	○	○	○	○
Il faut savoir faire des sacrifices pour réussir sa vie professionnelle.	○	○	○	○	○	○

7. Votre créativité

	1	2	3	4	5	NP
Je suis plus intuitif que logique.	○	○	○	○	○	○
Je suis curieux.	○	○	○	○	○	○
J'aime le changement.	○	○	○	○	○	○
Les problèmes un peu ardus excitent ma curiosité.	○	○	○	○	○	○
J'aime sortir des sentiers battus.	○	○	○	○	○	○

8. L’autodiscipline

	1	2	3	4	5	NP
Je m'efforce de gérer mon temps de sommeil.	○	○	○	○	○	○
Je refuse de remettre au lendemain ce que je peux faire le jour même.	○	○	○	○	○	○
Je m'efforce de corriger mes erreurs.	○	○	○	○	○	○
J'ai une forte capacité de travail si c'est pour mener à bien un projet.	○	○	○	○	○	○
Dans un projet, j'évalue toujours les conséquences de mes choix.	○	○	○	○	○	○

4. MA PERSONNALITÉ

Le test de personnalité permet d'évaluer cinq traits fondamentaux de votre personnalité suivant un modèle très utilisé en psychologie qui s'appuie sur vos goûts, vos habiletés et vos aspirations. Vous allez trouver un certain nombre de qualificatifs qui peuvent s'appliquer ou non à vous. Pour chaque affirmation, précisez le chiffre qui indique dans quelle mesure vous approuvez ou n'approuvez pas l'affirmation.

1 pour Désapprouve fortement
2 pour Désapprouve un peu
3 pour N'approuve ni ne désapprouve
4 pour Approuve un peu
5 pour Approuve fortement
6 pour Je ne souhaite pas répondre

Je me vois comme quelqu'un qui…	1	2	3	4	5	NP
… est réservé	○	○	○	○	○	○
… fait généralement confiance aux autres	○	○	○	○	○	○
… a tendance à être paresseux	○	○	○	○	○	○
… est « relaxe », détendu, gère bien le stress	○	○	○	○	○	○
… est peu intéressé par tout ce qui est artistique	○	○	○	○	○	○
… est social, extraverti	○	○	○	○	○	○
… a tendance à critiquer les autres	○	○	○	○	○	○
… travaille consciencieusement	○	○	○	○	○	○
… est facilement anxieux	○	○	○	○	○	○
… a une grande imagination	○	○	○	○	○	○

5. SIMULATION D'UN ENTRETIEN D'EMBAUCHE

Le questionnaire qui suit est une simulation d'entretien d'embauche pour un stage ou un emploi. Les questions sont des questions types que peut vous poser un recruteur. Répondez de la même façon que vous répondriez à un recruteur.

1. Êtes-vous en couple ?

- Oui
- Non
- Je ne souhaite pas répondre.

2. Êtes-vous marié(e) ?

- Oui
- Non
- Je ne souhaite pas répondre.

3. Avez-vous l'intention d'avoir des enfants ?

- Oui

- Non
- Je ne sais pas
- Je ne souhaite pas répondre.

4. Pour les affirmations suivantes, précisez le chiffre qui indique dans quelle mesure vous approuvez ou n'approuvez pas l'affirmation.

1 : Désapprouve fortement
2 : Désapprouve un peu
3 : N'approuve ni ne désapprouve
4 : Approuve
5 : Approuve fortement
NP : Je ne souhaite pas répondre

	1	2	3	4	5	NP
Entretenir plusieurs relations à la fois n'est pas possible pour moi.	○	○	○	○	○	○
Une relation affective stable aide à s'investir efficacement dans le travail.	○	○	○	○	○	○
Une vie affective inexistante permet d'être plus disponible professionnellement.	○	○	○	○	○	○
La fidélité en amour est une valeur importante pour moi.	○	○	○	○	○	○
Construire sa vie professionnelle est plus important que construire sa vie personnelle.	○	○	○	○	○	○
Réussir professionnellement rime avec réussite dans la sphère privée.	○	○	○	○	○	○
Adhérer à un syndicat est important quand on entre dans une entreprise.	○	○	○	○	○	○
Adhérer à un parti politique est important pour moi.	○	○	○	○	○	○

5. Faites-vous du sport ?

- Oui, dans un club de sport
- Oui, mais pas dans un club de sport
- Non, je n'ai jamais le temps
- Non, je n'aime pas le sport
- Je ne souhaite pas répondre.

6. Participez-vous à une association culturelle ?

- Oui, je suis même très actif
- Oui, mais pas autant que je le voudrais
- Non
- Je ne souhaite pas répondre.

7. Pensez-vous pouvoir travailler avec des collègues d'origine ethnique différente de la vôtre ?

- Oui
- Non
- Je ne souhaite pas répondre.

8. Accepteriez-vous de ne travailler qu'avec des femmes ?

- Oui
- Non
- Je ne souhaite pas répondre.

9. Accepteriez-vous de ne travailler qu'avec des hommes ?

- Oui
- Non
- Je ne souhaite pas répondre.

10. Accepteriez-vous de travailler avec une flexibilité totale des horaires ?

- Oui, si on me le demande.
- Oui, mais seulement si c'est nécessaire.
- Non, ma vie personnelle ne me le permets pas.
- Non, je ne le souhaite pas.
- Je ne souhaite pas répondre.

11. Accepteriez-vous de travailler 7 jours sur 7 ?

- Oui
- Oui, mais seulement sur une période limitée.
- Non
- Je ne souhaite pas répondre.

12. Accepteriez-vous de travailler sans être déclaré ?

- Oui, même à plein temps.
- Oui, mais pas à plein temps.
- Non, c'est trop incertain.
- Non, c'est trop dangereux.
- Je ne souhaite pas répondre.

13. Accepteriez-vous de travailler dans un local insalubre ou en travaux ?

- Oui, sans problème.
- Oui, si c'est temporaire
- Non, c'est trop dangereux.
- Je ne souhaite pas répondre.

15. Accepteriez-vous de travailler avec des homosexuels ?

- Oui
- Non
- Je ne souhaite pas répondre.

6.VIE PROFESSIONNELLE ET VIE PERSONNELLE

Dans un cadre professionnel, la frontière entre votre vie professionnelle et votre vie privée pourra être difficile à définir. Pour s'assurer de votre implication dans la vie de l'entreprise, un employeur peut en effet être amené à s'intéresser à vos activités personnelles. Par exemple l'usage de votre smartphone ou votre navigation sur Internet peuvent faire l'objet d'un contrôle de sa part. Les questions suivantes mesurent votre sensibilité à la collecte et au traitement de vos données personnelles. Répondez aux questions suivantes en indiquant sur une échelle de 1 à 5 si vous êtes ou non d'accord avec les affirmations suivantes :

1 : Pas du tout d'accord
2 : Plutôt pas d'accord
3 : Plutôt d'accord
4 : D'accord
5 : Tout à fait d'accord
NP : Je ne souhaite pas répondre

De façon générale, que pensez-vous de la collecte de données sur Internet ?

Cela me dérange lorsque des sites Internet me demandent des informations personnelles.
○1. ○2. ○3. ○4. ○5. ○NP.

Lorsque des sites Internet me demandent des informations personnelles, je réfléchis plutôt à deux fois avant de fournir ces informations.
○1. ○2. ○3. ○4. ○5. ○NP.

Cela me dérange de donner des informations personnelles à tant de sites Internet.
○1. ○2. ○3. ○4. ○5. ○NP.

Je suis préoccupé lorsque des sites Internet collectent trop d'informations personnelles à mon propos.
○1. ○2. ○3. ○4. ○5. ○NP.

Que pensez-vous des efforts que font les sites internets pour protéger les données personnelles des utilisateurs qu'elles possèdent ?

Les sites Internet devraient consacrer plus de temps et d'efforts à se protéger contre l'accès illégal aux informations personnelles.
○1. ○2. ○3. ○4. ○5. ○NP.

Les bases de données qui contiennent des informations personnelles devraient se protéger des accès illégaux, peu importe le coût.
○1. ○2. ○3. ○4. ○5. ○NP.

Les sites Web et autres entreprises devraient prendre davantage de mesures pour s'assurer que les hackers ne puissent pas accéder aux informations personnelles.
○1. ○2. ○3. ○4. ○5. ○NP.

Que pensez-vous des efforts que doivent faire les sites internet sur l'exactitude des données personnelles qu'ils possèdent ?

Toutes les informations reçues sur les sites Internet devraient être doublement vérifiées pour leur exactitude, peu importe le coût.
○1. ○2. ○3. ○4. ○5. ○NP.

Les sites Internet devraient prendre davantage de mesures pour s'assurer que les informations personnelles dans leurs fichiers sont exactes.
○1. ○2. ○3. ○4. ○5. ○NP.

Les sites Internet devraient avoir de meilleures procédures pour corriger les erreurs de saisies liées aux informations personnelles.
○1. ○2. ○3. ○4. ○5. ○NP.

Les sites Internet devraient consacrer plus de temps et d'efforts à vérifier l'exactitude des informations personnelles dans leurs bases de données.
○1. ○2. ○3. ○4. ○5. ○NP.

Que pensez-vous de l'usage potentiel que peuvent les entreprises de vos données personnelles ?

Les sites Internet ne devraient pas utiliser les informations personnelles à quelque fin que ce soit, sauf autorisation des individus qui fournissent ces informations.
○1. ○2. ○3. ○4. ○5. ○NP.

Lorsque les gens donnent des informations personnelles à un site Internet pour une raison déterminée, le site Internet ne devrait pas utiliser ces informations pour d'autres raisons.
○1. ○2. ○3. ○4. ○5. ○NP.

Les sites Internet ne devraient jamais vendre des informations personnelles qu'ils ont récoltées à d'autres sites.
○1. ○2. ○3. ○4. ○5. ○NP.

Les sites Internet ne devraient jamais partager des informations personnelles avec d'autres sites ou entreprises à moins que les individus qui fournissent ces informations aient donné leur autorisation.
○1. ○2. ○3. ○4. ○5. ○NP.

2. ENVIRONMENT OF EXPERIMENTATION

Figure 1. Environment of the Experimentation

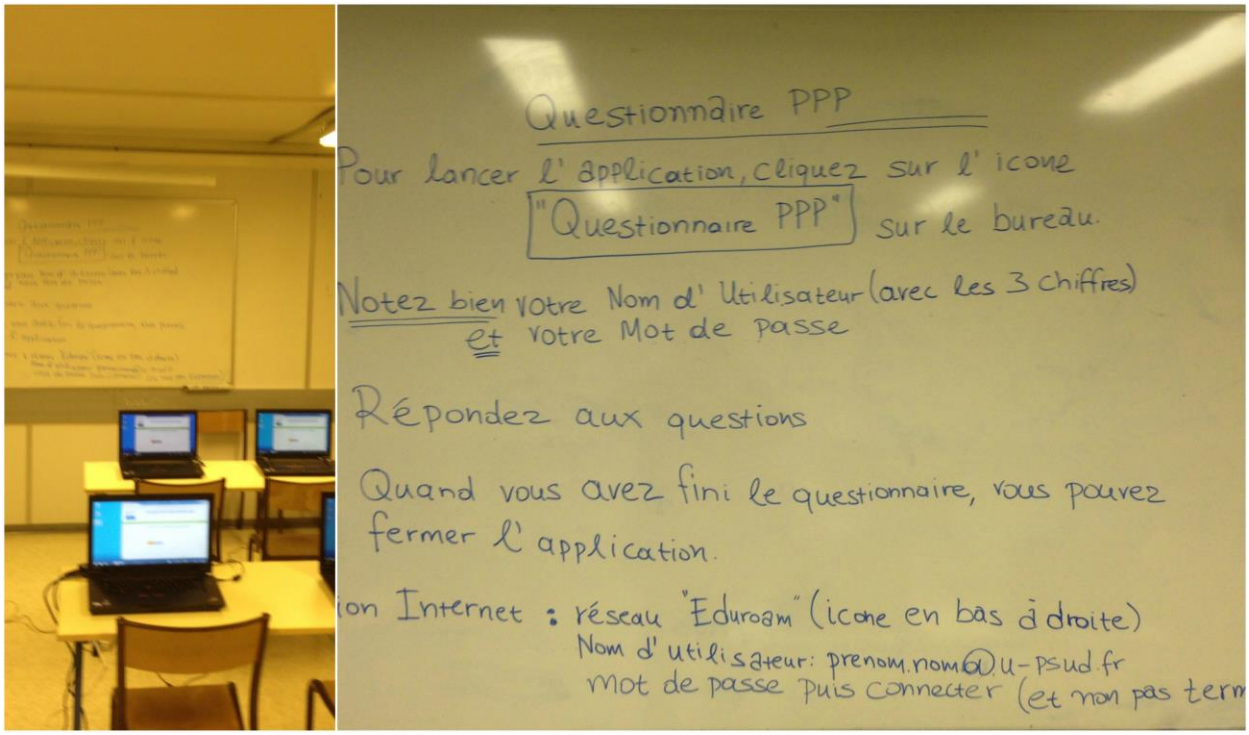

Figure 2. On-board Instructions

3. SENSITIVE QUESTIONNAIRE SURVEYS ONLINE

We present some screenshots of the website that we have started to implement, dedicated to the online questionnaire surveys.

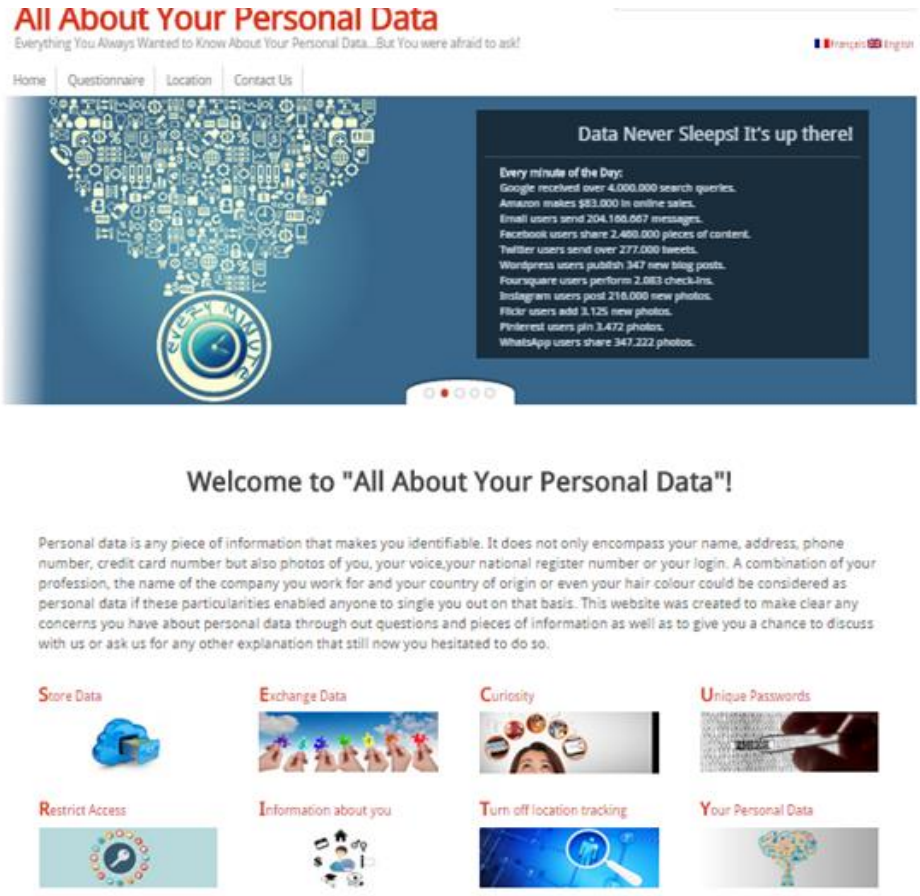

Figure 3. Initial Page of the website

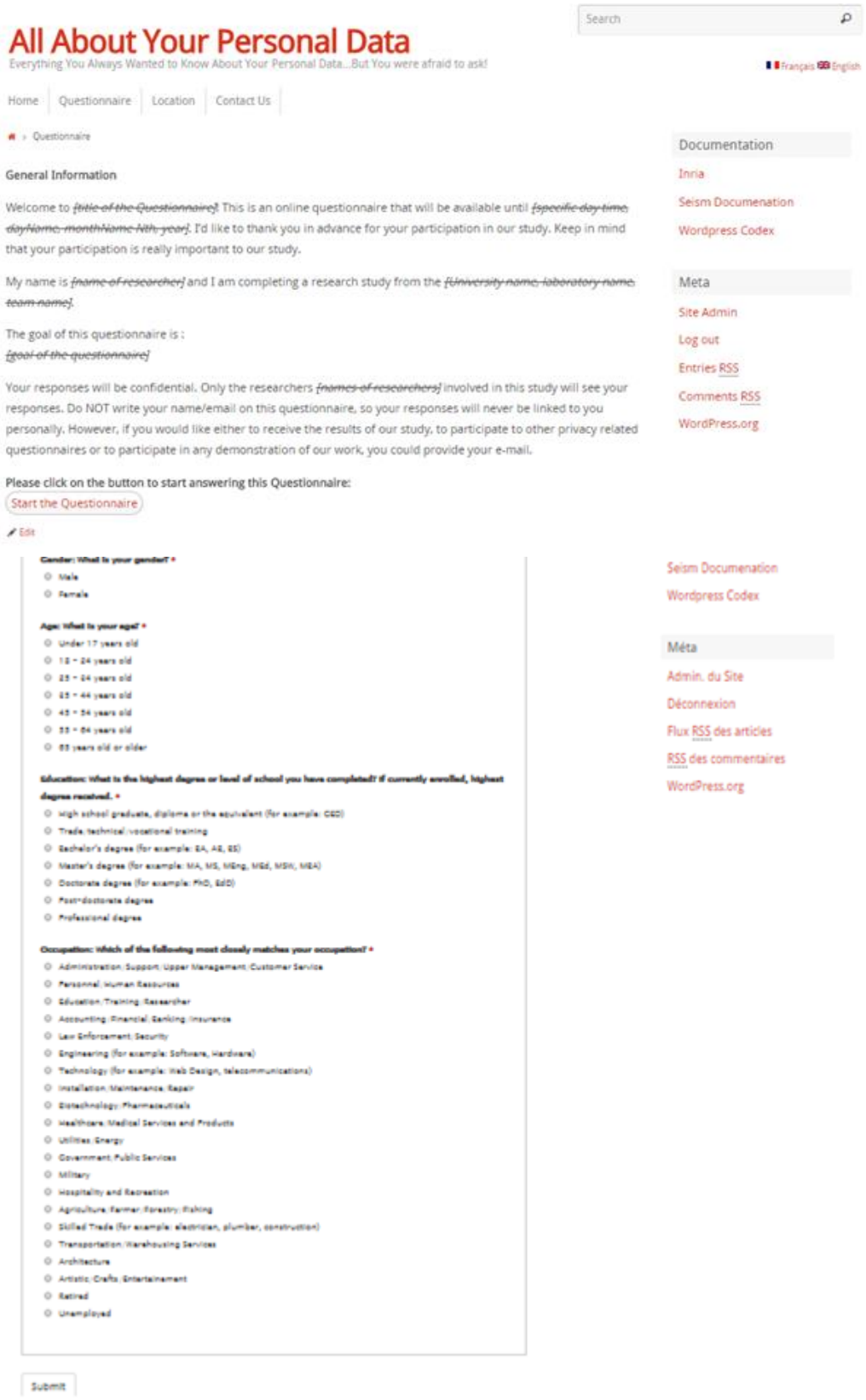

Figure 4. Questionnaire Survey

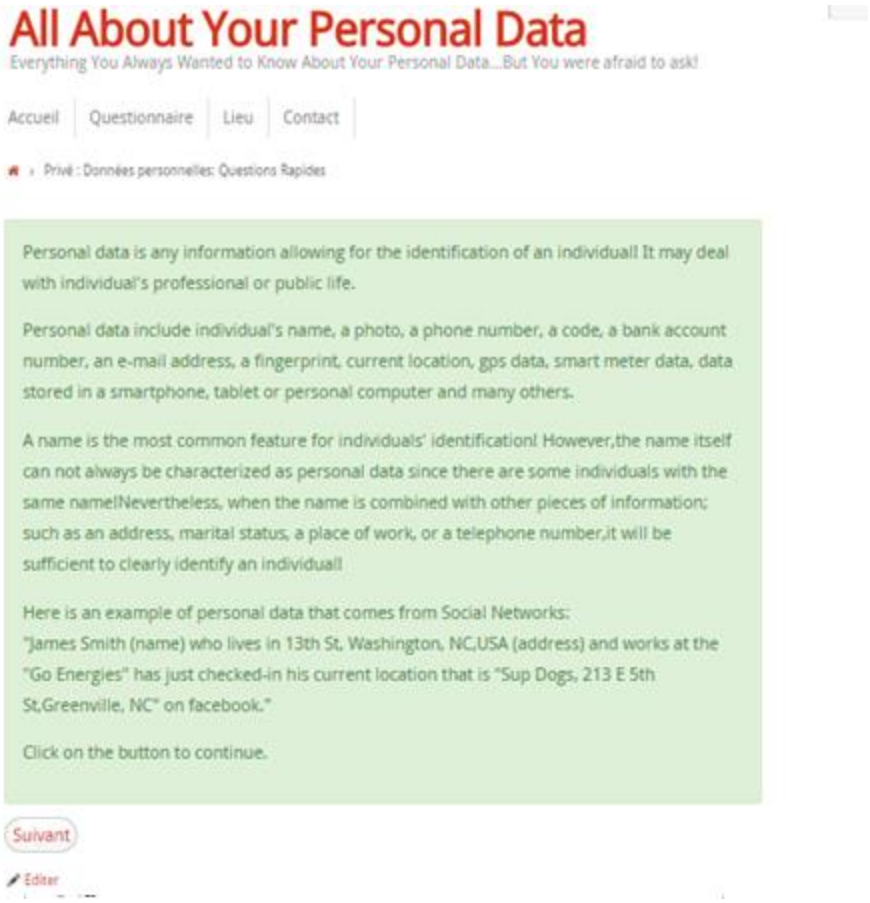

Figure 5. Information provided after answering the first set of questions