Extracting Access Control and Conflict Resolution Policies from European Data Protection Law

Kaniz Fatema, David Chadwick, Brendan Alsenoy

Extracting Access Control and Conflict Resolution Policies from European Data Protection Law

Kaniz Fatema[1], David W Chadwick[1], Brendan Van Alsenoy[2]

[1] School of Computing, University of Kent, Canterbury, UK
{k.fatema | d.w.chadwick}@kent.ac.uk
[2]Interdisciplinary Centre for Law and ICT (ICRI), K.U.Leuven, IBBT, Leuven, Belgium
Brendan.VanAlsenoy@law.kuleuven.be

Abstract. This paper presents the extraction of a legal access control policy and a conflict resolution policy from the EU Data Protection Directive [1]. These policies are installed in a multi-policy authorization infrastructure described in [2, 3]. A Legal Policy Decision Point (PDP) is constructed with a legal access control policy to provide automated decisions based on the relevant legal provisions. The legal conflict resolution policy is configured into a Master PDP to make sure that the legal access control policy gets priority over access control policies provided by other authorities i.e. the data subject, the data issuer and the data controller. We describe how clauses of the Directive are converted into access control rules based on attributes of the subject, action, resource and environment. There are currently some limitations in the conversion process, since the majority of provisions requires additional interpretation by humans. These provisions cannot be converted into deterministic rules for the PDP. Other provisions do allow for the extraction of PDP rules but need to be tailored to the application environment before they are configured into the Legal PDP.

Keywords: Legal PDP, Legal Access Control Policy, Conflict Resolution Policy, EU Data Protection Directive.

1 Introduction

Although there are a number of legal instruments aiming to protect the personal data of individuals [1, 4-7], the proper enforcement of these laws is often lacking. If the access control rules contained in these laws could be integrated in authorization infrastructures, this would arguably make the enforcement of data protection requirements more efficient and effective.

The use of policy based systems to protect personal data based on access control policies not new [8-11]. When developing a privacy preserving systems, it is important to keep in mind both the rights of the data subject as well as the legitimate interests of others. This balance of rights is critical and other researches, when designing policy based authorization systems to protect the privacy of personal data, have often overlooked this matter and focused primarily on the policy of the data subject [8, 9]. In comparison, we have attempted to support this balance by building multiple policy decision points (PDPs) into the authorization system so that all the

stakeholders can express their own independent policies[2]. The system is designed to include access control polices and conflict resolution polices [3] from different authors, possibly written in different policy languages (such as XACMLv2 [12], XACMLv3 [13], PERMIS [14], P3P [15] and so on) and these policies will be enforced in separate PDPs. Their decisions will then be combined by a Master PDP using the most appropriate conflict resolution policy.

For each item of personal data we consider policies may be written by up to four different types of authors: the law (the access control rules extracted from legislation will form a Legal PDP), the data issuer (e.g. for a degree certificate the university is the issuer, whilst for a personal diary the data subject is the issuer), the data subject (i.e. the individual to whom the data relates), and the data controller (i.e. the organization that is legally responsible for the personal data processing). When the controller's (or processor's) system receives a request to access a data item, it first retrieves all the policies related to the data item. The conflict resolution policies are prioritized in the order of law, issuer, data subject and controller. The policy which has the highest priority and is applicable to the current request is used by the Master PDP to resolve any conflicting decisions from the access control policies of the various authors.

The issuer's and subject's policies will always travel with the data as it moves across organizational boundaries. The controller's policy may or may not be transferred, since there might be a contract between the receiver (controller or processor) and sending controller which ensures that the receiver's policy shall be configured correctly. Alternatively, a subset of the sending controller's policy might be sent and merged with that of the receiver. The legal policy will not travel with the data if the data stays within the same jurisdiction (assuming the authorization system in the same jurisdiction has the same Legal PDP), but if the data is transferred to a foreign jurisdiction then it will be transferred. How foreign and local legal policies should interact and which should take precedence is still a matter for further study, since at the moment the rules for this are too complex to automate. We do not address this issue in the current paper, except that our Legal PDP can restrict the transfer of personal data to jurisdictions which do not have adequate legal protection unless the transfer is permitted by another lawful basis.

The main contribution of this paper is to present the rights of different groups of users to access any personal data as specified in the EU Data Protection Directive (EDPD), and to convert as many of these rights as possible into the rules of both an access control policy (held in a Legal PDP) and a conflict resolution policy (held in a Master PDP) so that automated and independent legally compliant decisions can be obtained. To our knowledge no other authorization system has such an integrated system. Having a separate Legal PDP and Master PDP is necessary to ensure that certain legal policies take priority over all other policies so that the rights provided by the law are not overridden.

The rest of this paper is structured as follows. Section 2 discusses related works; Section 3 describes the various steps of the methodology. Section 4 describes the validation test results and finally section 5 presents the limitations and conclusions.

2 Related work

The Enterprise Privacy Authorization Language (EPAL) [16] is a privacy language that helps policy writers to define terms, conditions and rules to protect customers' personal information. However, it does not provide the ability to express statements that need to be enforced by multiple policies or for the logical combination of policies. The NEURONA [17, 18] project developed a data protection application based on the Spanish data protection requirements that offer reports regarding the correct application of security measures to files containing personal data. If a file contains personal data but does not comply with the adequate level of security this is classified as an erroneous file by their ontology. They provide a semi-automated way to determine whether some aspects of the current state of a company's personal data files might not comply with the established set of regulations. Travis et al. [19-22] worked to automate the derivation of security requirements from regulations. They applied their method to the HIPAA regulations, but there is insufficient assessment of its applicability to other regulations [19]. Furthermore there are limitations in their method such as relying upon a specific format of regulatory texts, and relying on the analyst's skills [20]. In comparison, our work is based on an analysis and extraction of rules from the EDPD and converting these into executable policies so that automated decisions can be returned from a Master PDP.

3 Methodology

The methodology that we used to extract access control and conflict resolution rules from the EDPD [1] consists of seven procedural steps described below.

3.1 Step 1. Identifying provisions of the EDPD related to access control

The EDPD consists of seven chapters and 34 articles. We considered only those provisions which are directly related to access control. A provision is directly related to access control if it pertains directly to the access, collection, blocking or transfer of personal data.The general rules for the lawful processing of personal data are provided in chapter 2. The legitimate bases on which personal data can be processed are mentioned in Article 7. The legitimate bases on which sensitive personal data can be processed are mentioned in article 8. The information to be provided to the data subject while collecting or processing personal data is described in articles 10 and 11. Article 12 sets for the rights of the data subject with regards to the processing of his/her personal data, namely the rights of access, notice, rectification or blocking. The potential exemptions and restrictions to these rights are provided in article 13. The conditions under which personal data can be transferred to third countries are mentioned in articles 25 and 26; whilst article 28 describes the rights of supervisory authorities who is responsible for monitoring the application of their national data protection legislation.

3.2 Step 2. Extracting the Legal Access Control Policy

The provisions related to access control were examined one by one to assess whether they could (at least in part) be converted into rules that could be enforced automatically. Only the rules that are capable of giving an independent access control decision were kept i.e. the rules that are capable of saying who is allowed to perform which action on personal data under what conditions, or under what conditions personal data can be accessed. A provision was discarded if i) no access control rule can be extracted or ii) the extracted rule requires human judgment which cannot be easily translated into a deterministic rule, so that a fully automated enforcement is not possible. For example, article 6.1 (a) says "personal data must be processed fairly and lawfully" – this rule on its own is too vague to form an access control rule. Later in article 7 the criteria for making data processing legitimate are described, some of which can be converted into access control rules. For instance, article 7(a) states that "personal data may be processed only if the data subject has unambiguously given his consent". In our proposed system the data subject provides his/her consent in the form of the subject's privacy policy which says who may access his/her data for what purposes.

Article 6.2 states that "It shall be for the controller to ensure that paragraph 1 is complied with." This rule places responsibility on the controller to ensure that the EDPD is followed, but it does not form an access control rule itself. Article 8.2 (b) states that "processing is necessary for the purposes of carrying out the obligations and specific rights of the controller in the field of employment law insofar as it is authorized by national law providing for adequate safeguards" which is too complex to convert into an access control rule as it would for instance require encoding of all the employment laws. Article 12(b) states that "as appropriate the rectification, erasure or blocking of data the processing of which does not comply with the provisions of this Directive" is not possible to convert into an automated rule as it requires human judgement to evaluate whether the processing complies with the Directive. Article 12 (c) requires that third parties to whom data were disclosed be notified of any rectification, erasure or blocking carried out in compliance with article 12 (b). This rule is not feasible to present as an access control rule, but rather requires an update mechanism to satisfy the condition. After completing step 2 a total of 21 natural language rules were obtained**.**

The obligation to "Log the request" has been added only to the rules in which the actions may not have an immediate effect. For example, a data update request from the data subject may not take immediate effect if the controller needs to verify the accuracy or any other condition of the data.

Table 1. The Natural Language and Formalized Rules

1.	**Natural language rule:** If the requested purpose of processing does not match with the original purpose of data collection or is not for a historical purpose/statistical purpose / scientific purpose OR the validity time of data is before the requested time then deny the request. [From articles 6.1 (b) and 6.1 (e).] **Formalized elements for both ACR and CRR:** ***Subject:*** Anyone ***Resource:*** Personal Data ***Action:*** Read ***Condition: NOT (***RequestedPurpose = PurposeOfCollection OR RequestedPurpose = historical purpose OR RequestedPurpose = statistical purpose OR RequestedPurpose =scientific purpose) OR validity time earlier than the requested time. **Formalised elements for ACR:** ***Effect:*** Deny **Formalised elements for CRR:** ***Effect:*** Permit ***Obligation:*** use DCA=denyOverrides

2.	**Natural language rule:** A data subject can send data update requests for his/her personal data.[From article 6.1 (d) and article 12 (b).It is not possible to make an access control rule to verify which data are inaccurate or incompatible with regard to the purposes they were collected. However, this rule ensures that the data subject can send a data update request if he/she finds that the data is not accurate and the controller can either delete or update the data after judging the validity of the subject's request. The complete enforcement of this legal rule therefore requires human judgment.] **Formalized elements for both ACR and CRR:** ***Subject:*** Data Subject ***Resource:*** Personal Data ***Action:*** DataUpdateRequest **Formalised elements for ACR:** ***Effect:*** Permit ***Obligation:*** Log the request. **Formalised elements for CRR:** ***Effect:*** Permit ***Obligation:*** use DCA=grantOverrides
3.	**Natural language rule:** A data subject can submit a policy / update a policy for his/her personal data. [From articles 7 (a) and 8. 2 (a), by submitting and updating a policy the data subject can give or update or revoke his/her consent] **Formalized elements for both ACR and CRR:** ***Subject:*** Data Subject ***Resource:*** Personal Data ***Action:*** SubmitPolicy/ UpdatePolicy **Formalised elements for ACR:** ***Effect:*** Permit **Formalised elements for CRR:** ***Effect:*** Permit ***Obligation:*** use DCA=grantOverrides
4.	**Natural language rule:** If the purpose of data processing is performance of a contract and both the data subject and the requester are parties to the contract then grant access to the resource mentioned in the contract. [From article 7 (b) and 15.2 (a)] **Formalized elements for both ACR and CRR:** ***Subject:*** Anyone ***Resource:*** Personal Data ***Action:*** Read/Write ***Condition***: RequestedPurpose = performance of contract AND PartyOfContract= Data Subject AND PartyOfContract= Requester, AND requested resource = resource specified in contract **Formalised elements for ACR:*****Effect:*** Permit **Formalised elements for CRR:** ***Effect:*** Permit ***Obligation:*** use DCA=grantOverrides
5.	**Natural language rule:** Entities with a specific role (e.g. social security authority) can access a specific resource type (e.g., personal data related to pensions) and if the purpose is the performance of a task of public interest (e.g., social security administration) or an exercise of official authority. [From article 7 (e). The roles and resource types in the rule will need to be configured in light of the application and the national legislation.] **Formalized elements for both ACR and CRR:** ***Subject:*** Role X ***ResourceType:*** Y ***Action:*** Read ***Condition***:RequestedPurpose = performance of a task of public interest OR RequestedPurpose = exercise of official authority **Formalised elements for ACR:** ***Effect:*** Permit **Formalised elements for CRR:** ***Effect:*** Permit ***Obligation:*** use DCA=grantOverrides
6.	**Natural language rule:** Anyone with a Data Access Mandate can access the personal data. [From articles 7 (c), 8.2 (e) and 8.4]. See also section 3.5. **Formalized elements for both ACR and CRR:** ***Subject :*** Anyone ***Resource :*** Personal Data ***Action:*** Access ***Condition***: DataAccessMandate=true **Formalised elements for ACR:** ***Effect:*** Permit **Formalised elements for CRR:** ***Effect:*** Permit ***Obligation:*** use DCA=grantOverrides
7.	**Natural language rule**: A data subject can read his/her personal data if there is no legal objection within national legislation [From art. 12 and 13.1] ***a.*** **Formalized elements for both ACR and CRR:** ***Subject :*** Data Subject ***Resource:*** Personal Data ***Action:*** Read ***Condition:*** LegalObjection = true **Formalised elements for ACR:** ***Effect:*** Deny **Formalised elements for CRR:** ***Effect:*** Permit, ***Obligation:*** use DCA=denyOverrides. ***b.*** **Formalized elements for both ACR and CRR:** ***Subject:*** Data Subject ***Resource:*** Personal Data ***Action:*** Read **Formalised elements for ACR:** ***Effect:*** Grant **Formalised elements for CRR:** ***Effect:*** Permit, ***Obligation:*** use DCA=grantOverrides. [The rules will be executed in order, so that rule *b* will only be true if the condition of rules *a* is false or missing]
8.	**Natural language rule:** The treating Medical Professional can Read/Write medical data for the purpose of preventive medicine, medical diagnosis, prevention of care or treatment or the management of health care service. [From article 8.3] **Formalized elements for both ACR and CRR:** ***Subject:*** Medical Professional ***Resource :*** Medical Data ***Action:*** Read / Write ***Condition***: Medical Professional = a treating Medical Professional of the patient AND RequestedPurpose = medical diagnosis/ the provision of care and treatment / preventive

	medicine. **Formalised elements for ACR:** ***Effect:*** Permit **Formalised elements for CRR:** ***Effect:*** Permit, ***Obligation:*** use DCA=grantOverrides.
9.	**Natural language rule:** Medical professionals can BTG (break the glass) to medical data for purpose of medical diagnosis/ the provision of care and treatment / preventive medicine. [From articles 7 (d) and 8.2 (c). Break the glass is the ability to override access controls in case of emergency in order to gain access to data which is normally denied to the requester. This rule is an example of accessing personal data to save the vital interest of the data subject.] **Formalized elements for both ACR and CRR:** ***Subject:*** Medical Professional ***Resource:*** Medical Data ***Action:*** Read / Write ***Condition***: RequestedPurpose = medical diagnosis/ the provision of care and treatment / preventive medicine. **Formalised elements for ACR:** ***Effect:*** BTG **Formalised elements for CRR:** ***Effect:*** Permit, ***Obligation:*** use DCA=grantOverrides.
10.	**Natural language rule:** The data subject can send "Object to Processing" with an obligation to log the request. [From article 12 (b), 14 (b) and 15.1] **Formalized elements for both ACR and CRR:** ***Subject:*** Data Subject ***Resource:*** Personal Data ***Action:*** Object to Processing **Formalised elements for ACR:** ***Effect:*** permit, ***Obligation:*** Log the request **Formalised elements for CRR:** ***Effect:*** Permit, ***Obligation:*** use DCA=grantOverrides.
11.	**Natural language rule:** Medical Professional can BTG to transfer medical data to a non EU/EEA country not having an adequate level of protection**.** [From article 26.1 (e)] **Formalized elements for both ACR and CRR:** ***Subject:*** Medical Professional ***Resource:*** Medical Data ***Action:*** Transfer to Country=X ***Condition:*** value of X = non EU/EEA country not having adequate level of protection **Formalised elements for ACR:** ***Effect:*** BTG **Formalised elements for CRR:** ***Effect:*** Permit**,** ***Obligation:*** **use DCA=**grantOverrides**.**
12.	**Natural language rule:** Anyone can transfer personal data from public register. [From article 26.1 (f)] **Formalized elements for both ACR and CRR:** ***Subject:***Anyone ***Resource:*** personal data in public register ***Action:*** Transfer to Country=X ***Condition:*** value of X = one of the non EU/EEA country not having adequate level of protection **Formalised elements for ACR:** ***Effect:*** Permit **Formalised elements for CRR:** ***Effect:*** Permit, ***Obligation:*** use DCA=grantOverrides.
13.	**Natural language rule:** Personal data can be transferred to a non EU/EEA country not having an adequate level of protection if the subject unambiguously consents to the transfer OR if the purpose is performance of contract and the parties of contract are data subject and controller OR the parties of contract are controller and third party and contract's beneficiary is the data subject OR there is a transfer mandate; otherwise deny the transfer." [from article 26.1 (a), (b), (c), (d), 26.2 and 25.4] ***a.*** **Formalized elements for both ACR and CRR:** ***Subject:*** Anyone ***Resource:*** Personal data ***Action:*** transfer to country=X ***Condition:*** value of X = (one of the non EU countries OR countries not having adequate level of protection) AND(value of SubjectConsentsToTransferTo= ID of requester OR (RequestedPurpose = performance of a contract AND ((PartyOfContract=Data Subject AND PartyOfContract= controller) OR (PartyOfContract=controller AND SubjectOfContract=dataSubject AND BeneficiaryOfContract=Data Subject))) OR DataTransferMandate= true) **Formalised elements for ACR:** ***Effect:*** Permit **Formalised elements for CRR:** ***Effect:*** Permit, ***Obligation:*** use DCA=grantOverrides. ***b.*** **Formalized elements for both ACR and CRR:** ***Subject:*** Anyone ***Resource:*** Personal data ***Action:*** transfer to country = X ***Condition:*** value of to "country X" ≠ one of the countries in the list of allowed countries **Formalised elements for ACR:** ***Effect:*** Deny **Formalised elements for CRR:** ***Effect:*** Permit, ***Obligation:*** use DCA=denyOverrides. [The rules are executed in order, so the rule b will only be executed when the conditions of a is not satisfied]
14.	**Natural language rule:** The Supervisory Authority can access and collect personal data for the performance of supervisory duties. [From article 28.3] **Formalized elements for both ACR and CRR:** ***Subject:*** Supervisory Authority ***Resource:*** Personal data ***Action:*** Access/ Collect ***Condition***: RequestedPurpose= performance of supervisory duties **Formalised elements for ACR:** ***Effect:*** Permit **Formalised elements for CRR:** ***Effect:*** Permit, ***Obligation:*** use DCA**=**grantOverrides.

15.	**Natural language rule:** The Supervisory Authority can order the blocking/erasing /destruction of data, or impose a temporary ban on the processing or impose a definitive ban on processing. [From article 28.3] **Formalized elements for both ACR and CRR:** ***Subject*** Supervisory Authority ***Resource:*** Personal data ***Action:*** Order to block/ Order to erase / Order to destruct / Impose temporary ban/ impose definitive ban ***Condition***: RequestedPurpose= performance of supervisory duties **Formalised elements for ACR:** ***Effect:*** Permit ***Obligation:*** LogTheOrder **Formalised elements for CRR:** ***Effect:*** Permit, ***Obligation:*** use DCA=grantOverrides.

3.3 Step 3. Refining the Access Control Rules

The natural language rules from step 2 were refined during this step by eliminating redundancy and duplication and by joining the rules together (if possible) to reduce their number. For example, the rule to deny an access if the purposes don't match with the original purposes of collection and the rule of denying an access if the request is made after the validity time is over are combined to form one rule. The two rules that allow/deny access to personal data by the data subject are combined together and the five rules that allow/deny the transfer of personal data to another country are combined to form a single rule. This left fifteen rules which are shown in Table 1. The order of rules is also determined at this step. For example, while combining the rules to allow/deny access to the data subject, the more specific rules that restrict the access are placed before the rule that grants access to the data subject. This ordering makes sure that the data subject can get access only when the conditions for which restrictions apply are false.

3.4 Step 4. Formalizing the Access Control Rules

In this step each of the natural language rules was formalized in the form of an Access Control Rule (ACR) and a Conflict Resolution Rule (CRR) as follows:

ACR – Subject: (who) Resource: (which data item) Action: (what action) Condition: (under what conditions) Effect: (Permit, Deny or BTG[1]) Optional Obligation: (subject to these actions being carried out);
CRR – Subject: (who) Resource: (which data item) Action: (what action) Condition: (under what conditions) Effect: (Permit) Obligation: (Decision Combining Algorithm (DCA) to be returned).

Each legal ACR rule has a matching CRR to make sure the legal rule gets precedence over any other author's rules. The difference between the CRR and its corresponding ACR is that the effect of the CRR is always Permit and the obligational ways returns the DCA that is applicable. If an ACR has an effect of Deny, the corresponding DCA

[1] Break the Glass means that the requester is not normally allowed to access the resource, but if they deem it to be an emergency situation, they can override the access controls and access the resource, in the full knowledge that their actions will be monitored and they will be answerable for them to a higher authority.

is denyOverrides and if an ACR has an effect of Permit the corresponding DCA is grantOverrides. If the ACR has an effect of BTG, the CRR's DCA is grantOverrides, since a grant from another PDP should not require the requester to first break the glass before gaining access. The formalization of ACRs and CRRs is provided in Table1.

3.5 Step 5. Attribute determination

For automated execution of these rules in an attribute based access control (ABAC) system [23] we need to determine the attributes of each of the elements in the rules. Four different types of attribute are used in constructing the policy rules. 1. Subject attributes 2. Action attributes 3. Resource attributes 4. Environment attributes. Subject attributes identify the users who are to be granted or denied access. Action attributes describe the action that is being controlled. Resource attributes describe the protected resource. In this case they are the metadata which describes the personal data being protected, and comprise attributes such as: resource type, data owner/issuer, data subject, date of creation etc. Environmental attributes describe the context in which the rule applies, such as time of day, location etc. These four types of attribute are also used to describe a user's request to access a resource, and are passed to the PDP in the request context by the application. PDPs compare the attributes of the user's request with those of the rules to determine whether access should be granted or not. We assume that the application is capable of storing the resource and its metadata securely, and retrieving the resource attributes and passing them to the Master PDP in the request context when a request for accessing that resource is received.

- **The data subject** is determined based on his/her set of identifying attributes (such as name and address, e-mail address, NI number, NHS Number etc.) given at the time the personal data is submitted or during the registration of the subject with the controller for a service. These identifying attributes become part of the resource's metadata. The Legal PDP checks if these identifying attributes match those of the requester (passed with the request context as subject attributes) to determine whether the requester is the data subject or not for the requested resource. In the current implementation the following sets of uniquely identifying attributes are used:{name and address}, {e-mail address}, or {NHS Number}, but these sets are configurable and can be changed and extended as needed by the application. The data subject should be able to choose any of these set of attributes to identify her/himself.

- **ResourceType** is a resource attribute that holds the type of the data, such as medical record, and is placed as a metadata of the resource by the issuer of the data. Only the issuer of the data can modify the ResourceType of that data. An ontology is needed to classify the different types of personal data, and an ontology mapping server (e.g. as described in [3]) may be used to hold the data classification ontology and be able to determine whether a resource type is a type of personal data or not. The ontology server may also determine the relationship among these data types. For example, all the medical data types are subclasses of personal data. The rules for personal data will therefore be applicable to medical data but not the other way around.

- **PurposeOfCollection** (mentioned in rule 1 of Table 1) is another resource metadata attribute that states the set of purposes for which the data was collected from the data subject. It is set by the application when the data is first collected from the data subject. The Legal PDP matches this set with the RequestedPurpose(s) stated by the requester in the request context.

- **ValidityTime** (mentioned in rule 1 of Table 1) is another resource metadata attribute collected from the issuer or data subject. A default value can also be set by the controller if the issuer or data subject does not provide a value for it. The controller will need to mention the default validity time of the data when collecting it. The Legal PDP matches the time of the access request (passed as an environment attribute of the request context) with the ValidityTime of the requested data.

- **Treating medical professional** (mentioned in rule 8 of Table 1) is identified by an identifying attribute stored in the medical record of the patient (as a part of the metadata). The value of this attribute must match that of the equivalent attribute of the requester, in order for the requester to be identified as the treating medical professional. The name of this attribute is configurable in the legal policy.

- **Medical Professional/ Supervisory Authority** are Role attributes (mentioned in rules 5, 8, 9, 14 and 15 of Table 1) provided by trusted Attribute Authorities. Who are the trusted authorities for which roles depends upon the application, and these are configurable values in the legal policy.

- **LegalObjection** (mentioned in the rule 7 of Table 1) is a Boolean attribute of a resource (metadata) which are used to flag personal data which is not accessible to the data subject because of national legislation which contains an exception to the data subject's right of access (e.g., a doctor may have the ability to invoke a therapeutic exception to prevent the patient from accessing certain information). These attributes can only be issued by the designated (trusted) authorities.

- **Data Access Mandate/ Data Transfer Mandate** (mentioned in rules 6 and 13 of Table 1) are credentials which can only be obtained by a requester following the appropriate legal procedure. Conceptually these are treated as subject attributes in the policy, so that if a requester possesses the appropriate mandate attribute he/she inherits the permissions assigned to the mandate (the Data Access Mandate is assigned for allowing access to personal data and the Data Transfer Mandate is assigned for allowing the transfer of personal data). These legal mandates are issued by various trusted Attribute Authorities and both the trusted authorities and mandate types are configurable to suit the application. The requester (or the Attribute Authority) presents the Mandate to the application which will either verify it using a Credential Validation Service and pass the valid attribute to the PDP as a subject attribute, or pass it to the PDP as a subject credential for the latter to verify.

- **PartyOfContract and SubjectOfContract** (mentioned in rules 4 and 13 of Table 1) are attributes of a contract. A contract is hypothesised to be a digitally signed XML document which has an element called PartyOfContract containing the IDs of

the people who are parties to the contract, and an element called SubjectOfContract containing the ID of the data subject. When a requester wants to access a subject's personal data for a purpose related to performance of the contract s/he will present the contract to the system. The ID of the data subject will be matched with the attributes in the contract. If both the requester and the subject are parties of the contract then access will be granted. If both the requester and the controller are parties of the contract and the data subject is the subject of the contract, then access will be granted. For validating contracts, a new trusted component called a Contract Validation Service (ConVS) is added to the system. To be able to access a data item based on a contract the requested data item should be mentioned in the contract. Therefore, in the contract along with the ID of the data subject (as a SubjectOfContract) the ResourceType is mentioned. If the requested resource's ResourceType and ID of the Data Subject mentioned in the metadata of the resource do not match with the ResourceType and the ID of the SubjectOfContract mentioned in the contract the access will not be granted based on contract.

- **SubjectConsentsToTransferTo** (mentioned in rule 13 of Table 1)is an environment attribute set by the application to the ID of the requester when a data subject consents to transfer his/her personal data to a requester. A requester can send a request for consent (via the application) to the data subject for transferring his/her personal data. If the data subject agrees to the transfer s/he can give his/her consent via the application (e.g. by clicking a button or ticking a box). This consent will be stored by the application and when the requester requests the data this consent (in the form of SubjectConsentsToTransferTo environment attribute) is appended to the request context by the application.

3.6 Steps 6 and 7. Implementation and Validation

The legal ACP (Access Control Policy) and legal CRP (Conflict Resolution Policy) containing the ACRs and CRRs have been converted into machine executable policies using both the XACML and PERMIS policy languages. For the construction of the CRP the format presented in [3] is followed. All the CRPs are inserted in order into a Master PDP with the precedence of law, issuer, data subject and controller. When the first executed condition is satisfied by a request context a Grant decision is returned along with an obligation to use the enclosed DCA which is used to combine the various decisions returned by the Legal, issuer's, data subject's and controller's PDPs. The current implementation of the system can be downloaded from [24].

The authorization server is initialized with a configured Legal PDP, controller's PDP and Master PDP (containing a set of CRPs). Data subjects' access control policies are dynamically inserted into the system as sticky policies when the subjects' personal data is first received. Likewise a data issuer's access control policy may also be received as a sticky policy along with the data, unless the data controller or data subject is also the data issuer. In the former case the issuer's policy will be created when the data is created locally, in the latter case there won't be an issuer's policy. On the receipt of an authorization decision request, the authorization server retrieves the issuer's and the subject's sticky policies based on the requested resource id. A

limitation of the current implementation is that it cannot dynamically process sticky conflict resolution policies.

Validation is performed by loading the XML implementation of legal policies into two PDPs – the Master PDP which determines and enforces the applicable DCA for each access request, and the Legal PDP which returns the legal decision for the request. These PDPs are then combined with the subject's, issuer's and controller's PDPs as appropriate. A set of test cases were generated in which various parties make different requests to access different personal data items. Comparison is done to see if the machine generated decisions and the human computed (correct) decisions are the same.

4 Validation Test Results

In order to validate the system two different sets of scenarios were developed: medical and employment. Here we only describe a subset of the medical scenario. Patient Mr. M registers with the Kent Health Centre and completes a registration form. He also gives his policy for access to his data, namely: Researchers are allowed to view my medical data for the purpose of medical research if the data is anonymized. His conflict resolution policy is: if a request is for my personal data DCA=denyOverrides. In addition the controller's policies say: 1. Administrative Officers can read and write administrative data (such as the contact information of patients, and which doctor is treating which patient etc.) but can't access the medical data. 2. Financial Officers can read the billing and payment information but can't read the medical data or administrative data. 3. Medical professionals can't access the billing and payment information or the patient's financial information. The controller's CRR says, for the request of any data the DCA is grantOverrides. Each resource has a unique RID and each resource consists of data and meta data. When creating data about a subject, the issuer mentions his ID (it can be a Role instead) for identifying himself and also mentions his access control policy saying 1. Only the issuer of the data can change the metadata such as the ResourceType. 2. Medical Professional can issue LegalObjection if there is a therapeutic exception which means that if the Medical Professional thinks that seeing the medical record may cause any mental or physical harm to the patient then s/he can issue a LegalObjection to stop the data subject from seeing the record. An inbuilt issuer CRR says that if the requester is the issuer and the resource=metadata then DCA=grantOverrides. An issuer is identified by the ID s/he provides when s/he first issues the data.

Mr. M goes for treatment to Dr. D at the Kent Health Center. When Dr. D tries to access Mr. M's medical data the Legal CRP returns grantOverrides and the Legal PDP returns Grant (based on rule 8 of Table 1) and so the final decision is Grant. Mr. M then has an X-ray. Dr. D enters the preliminary results into the Mr. M's record, and again the Legal PDP grants access as before. Dr. D decides to refer the patient to a lung specialist Dr. S at a London Hospital. Mr. M goes to Dr. S at the London hospital, who tries to access Mr. M's record at the Kent Health Center. The Legal CRP returns grantOverrides and the Legal PDP returns decision BTG (based on rule 9) and the other PDPs return notApplicable as they don't have any policy regarding

the request. BTG is the final decision which is meant to be used for exceptional situations only. Dr. S suggests that Mr. M changes his policy at the Kent Health Center to allow Dr. S to access his medical data in future without having to break the glass. Mr. M requests the Kent Health Center to update his access control policy. The Legal CRP returns grantOverrides as the data subject is requesting to update his policy and the Legal PDP returns Grant (based on rule 3), so Mr. M changes his policy to allow Dr. S at the London hospital to access his medical record. The next time Dr. S tries to access the medical record the Legal CRP returns grantOverrides and the Legal PDP returns BTG as before, while Mr. M's PDP returns Grant; so the final decision is Grant and Dr. S can read the medical record straight way.

Mr. M is suspected of having an illness of which Dr. S suspects that if Mr. M knows this it would be harmful for him to know that at the moment, so he issues LegalObjection=True for the metadata of this medical data to protect the data subject from a serious harm. This is granted due to the issuer ACR and CRR. Now Mr. M requests to view all his medical data. The application goes through each record asking if Mr. M can read the record. The Metadata of this specific medical record of Mr. M will have. The Legal CRP returns denyOverrides and the Legal PDP returns decision Deny for this record (based on rule7.a) and Grant for the other records (based on rule 7.b), so Mr. M does not learn of this specific illness.

An administrative officer tries to read the Medical record of Mr. M. The Legal CRP and Issuer CRP do not have any CRR matching the request context so the DataSubject's CRP returns denyOverrides. The Legal, issuer and Mr. M's PDPs return notApplicable, whilst the controller's PDP returns Deny so the final decision is Deny.

A researcher wants to access the medical data of Mr. M for medical research at the Kent Health Centre. The DCA is denyOverrides by the DataSubject's CRP. The Legal, issuer and controller PDPs return notApplicable, Mr. M's PDP returns Grant with an obligation to anonymise the data. If the obligation can be fulfilled by the application then a Grant decision (according to the algorithm of denyOverrides [3]) is returned along with the anonymised data. If the obligation can't be fulfilled a Deny decision is returned.

5 Conclusion and future work

We have presented a system that incorporates a Legal PDP and legal conflict resolution rules into an authorization server to enforce several of the rights and obligations outlined in the EDPD. The main advantage of having a separate Legal PDP is that it can automatically enforce certain legal provisions and allows administrators to see what these rules are. Having separate conflict resolution rules enforced by a Master PDP ensures that no other PDP can override the decisions of the Legal PDP. However, both the current design and implementation have their limitations. The Legal PDP does not completely capture all the legal constrains due to the complexity and nature of some of its rules. Some conditions are extremely complex to automate and some decisions are highly dependent on human judgment and/or intervention, so these cannot be automated. For instance, if a data subject

exercises his right of deletion or blocking of data on the basis that the data was obtained unlawfully, human intervention is necessary in order to determine whether this is in fact the case. Furthermore, how foreign and local legal policies should interact and which should take precedence is still a matter for further study, since at the moment the rules for this are too complex to automate. The proposed approach is therefore an initial proof of concept only. Several requirements may be far more complex in practice than are presented here, due to the divergences of national or sector specific laws from the European Directive. In a future contribution we aim to develop rules which can better accommodate this greater complexity.

References

1. Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data.
2. Chadwick, D. W. and Fatema, K.:An advanced policy based authorisation infrastructure. In Proceedings of the 5th ACM workshop on Digital identity management (DIM '09). ACM, New York, NY, USA, 2009.
3. Fatema, K., Chadwick, D. W., Lievens, S.:A Multi-privacy Policy Enforcement System. In: Privacy and Identity 2010, IFIP AICT 352, 2011, pp. 297–310.
4. OECD, Privacy and Personal Data Control, http://www.oecd.org/dataoecd/30/32/37626097.pdf
5. Health Information Privacy, HIPAA 1996 privacy and Security Rules, http://www.hhs.gov/ocr/privacy/
6. Protection of personal information in the private sector,http://www2.parl.gc.ca/HousePublications/Publication.aspx?pub=bill&doc=C-6&parl=36&ses=2&language=E&File=32#4
7. Australian Govt. ComLaw, Privacy Act 1988,http://www.comlaw.gov.au/Series/C2004A03712
8. Karjoth, G., Schunter, M., Waidner, M.: Privacy-enabled services for enterprises. In: 13th International Workshop on Database and Expert Systems Applications, pp. 483-- 487. IEEE Computer Society, Washington. DC (2002)
9. Mont, M. C.: Dealing with Privacy Obligations: Important Aspects and Technical Approaches. In: International conference on trust and privacy in digital business No1, Zaragoza (2004)
10. Ardagna, C. A.,Bussard, L., Vimercati, S. D. C., Neven, G. Paraboschi, S.,Pedrini, E.,Preiss, F-S.,Raggett, D.,Samarati,P.,Trabelsi, S.,Verdicchio, M.: PrimeLife Policy Language. In: project's position paper at W3C Workshop on Access Control Application Scenarios, November, 2009.
11. Trabelsi, S. Njeh, A., Bussard, L. and Neven, G.: PPL Engine: A Symmetric Architecture for Privacy Policy Handling. In:position paper at W3C Workshop on Privacy and data usage control, October, 2010.
12. OASIS XACML 2.0. eXtensible Access Control Markup Language (XACML) Version 2.0, Oct, 2005, http://www.oasisopen.org/committees/tc_home.php?wg_abbrev=xacml#XACML20.
13. OASIS XACML 3.0. eXtensible Access Control Markup Language (XACML) Version 3.0, 16 April, 2009, http://docs.oasisopen.org/xacml/3.0/xacml-3.0-core-spec-en.html
14. Chadwick, D., Zhao, G., Otenko, S., Laborde, R., Su, L. and Nguyen, T. A.:PERMIS:

a modular authorization infrastructure. In: Concurrency And Computation: Practice And Experience, vol 20, issue 11, pp 1341-1357.(2008)
15. W3C: The Platform for Privacy Preferences 1.0 (P3P 1.0), Technical Report 2002.
16. Ashley, S.H.P., Karjoth, G., Powers, C., and Schunter, M. "Enterprise Privacy Authorization Language (EPAL 1.2)," presented at W3C Member Submission, 2003.
17. Casellas, N., Mozos, M. R. D. L, Casanovas, P. : Ontology-Enhanced Legal Decision-Support Tools: The NEURONA Data Protection Compliance Application, http://www.lefis.org/app/eportfolio/artefact/file/download.php?file=584&view=61
18. Casellas, N.,Nieto, J-E, Merono, A., Roig, A., Torralba, S., Reyes, M., Casanovas, P.: Ontology Semantics for Data Privacy Compliance: The NEURONA Project,www.aaai.org/ocs/index.php/SSS/SSS10/paper/download/1071/1476
19. Breaux, T. D., Antón, A. I.: Analyzing Regulatory Rules for Privacy and Security Requirements. In:IEEE Transactions on Software Engineering, Special Issue on Software Engineering for Secure Systems *(*IEEE TSE*)*, 34(1):5-20, January/February 2008.
20. Breaux, T. D., Antón, A. I.: A Systematic Method for Acquiring Regulatory Requirements: A Frame-Based Approach. In: Proc. 6th International Workshop on Requirements for High Assurance Systems *(*RHAS-6*)*, Delhi, India, Sep. 2007.
21. Breaux, T. D., Antón, A. I. Antón.: Analyzing Goal Semantics for Rights, Permissions and Obligations. In: Proc. IEEE 13th International Requirements Engineering Conference (RE'05), Paris, France pp. 177-186, Aug. 2005.
22. Kiyavitskaya, N., Zeni, N., Breaux, T.D,, Antón, A.I., Cordy, J.R., Mich, L., Mylopoulos, J.: Automating the Extraction of Rights and Obligations for Regulatory Compliance. In: Proc. 27th International Conference on Conceptual Modelling (*ER'08*), Barcelona, Spain, pp. 154-168, Oct. 2008
23. ITU-T Rec X.812 (1995) | ISO/IEC 10181-3:1996 “Security Frameworks for open systems: Access control framework”
PERMIS, Standalone authorization Server,
http://sec.cs.kent.ac.uk/permis/downloads/Level3/standalone.shtml