

Challenges and Potential Solutions for Secure and Efficient Knowledge Leveraging in Coalitions

Petros Belsis

Laboratory of Information & Communication Systems Security (*Info-Sec-Lab*),
Department of Information & Communication Systems Engineering,
University of the Aegean, Samos, GR-83200, Greece
E-mails: pbelsis@aegean.gr URL: www.icsd.aegean.gr/pbelsis

Abstract

Dynamic coalitions are often formed to facilitate the sharing of knowledge between different organizations that have to collaborate on the grounds of a common purpose (such as emergency incident responding). In such a versatile and dynamic environment - where participant domains may leave or join at any time- the ability to retrieve and disseminate knowledge efficiently and rapidly is a task of extreme significance. Secure management of resources on the other hand is a non-trivial process. In this paper, we present the main challenges in coalition management and provide a framework for efficient classification and retrieval of knowledge assets. We also describe a flexible security framework for coalition management, as well as an optimization effort based on soft constraints that allows the reduction of the system's administrative overhead.

Keywords: Coalitions, Security Policies, Soft Constraints.

1. Introduction

Under modern multinational and multidisciplinary business models, there often exists the necessity for different organizations to form coalitions. In these coalitions, resources data and applications are shared between the participant domains. Such federations are highly desirable in academic, governmental, medical etc. alliances, or may be formed under the urge to respond to emergency situations (i.e. physical disasters, terrorist attacks prevention). We can identify two main challenges under these circumstances: first, to determine a way to retrieve in a fast and accurate manner the necessary information and disseminate it, while minimizing - whenever possible - network resources consumption; second, to provide access to resources only to those users who are authorized to do so. The inherent dynamic nature of the coalition maximizes the problems related with its security management. Participant domains may join or leave at any moment, while the global policy that rules the coalition has to be compliant to the restrictions imposed by individual policies. The formation and management of the coalition is a difficult, resource consuming and error-prone process. This is due to the large number of potential participants, the number of shared applications and resources, as well as due to the difficulties in merging diverse policies. It is therefore essential to provide support by means of automated tools and flexible methods [GKK01].

Previous research work has pinpointed the main challenges related with the specific problem. While looking at the problem from a single-domain perspective, efficient ways to classify the knowledge sources are necessary [BON02][BGM04][BGS05]. Also, actions over the shared resources should be compliant to the individual policies [SJB05][GRI06]. Looking at the problem from a broader, multi-domain perspective, the main challenges include: to retrieve the most relevant knowledge sources to a user query; to minimize necessary network bandwidth when directing the query to different domains; to establish a way to merge the different policies. This merging should be compliant to two fundamental security principles [GQ94][SJB05]:

- The principle of autonomy: if an access is permitted within an individual system, it must also be permitted under secure interoperation.
- The principle of security: if an access is not permitted within an individual system, it must not be permitted under secure interoperation.

The contribution of this paper is twofold. First we provide techniques for efficient knowledge classification and retrieval and introduce a solution for its efficient dissemination in a distributed federated environment. Second, we propose a solution for coalition oriented access control management. We present the generic components of our architecture, modules of which have been also presented in [BGM05] [BGK05] [MPB05a] [MPB05b] and we describe a technique that optimizes security management by reducing the administrative overhead.

The rest of the paper is organized as follows: Section 2 discusses the main choices for efficient knowledge assets filtering, while Section 3 describes solutions to provide efficient querying in a multi-domain environment. Section 4 introduces access control solutions for coalitions based on policy mappings while the basic principles of our authorization module are also explained. Section 5 introduces a formalism to represent access control problems in dynamic coalitions, utilizing soft constraints. Section 6 briefly reviews related work and section 7 concludes the paper.

2. Efficient knowledge assets classification

Our work focuses on enabling efficient retrieval, classification and dissemination of knowledge assets in distributed environments. In our approach, we first consider solutions for classification of documents according to predefined criteria. In this respect we define a solution that allows efficient querying of different domains and present an initial solution for access control interoperation. The latter will then be optimized and elaborated with the introduction of an appropriate formalism.

In the following section we sketch a solution that allows classification of documents to two or more classes, according to the document's content relevancy to one of the pre-specified categories. The classes can be specified by the user's interests. The requirement is that there is a small, manually created training set that fits the classification criteria imposed by the user. The main idea of the system is that every document can be considered as either containing several characteristics or not. The predefined characteristics are recorded in a document matrix, and accordingly the system examines the document for the presence of some, all or none of them.

2.1 Selecting features to facilitate classification

We consider that we have a set of documents (documents of the manually classified training set, as well as unclassified documents) which can be represented as vectors of binary features: $e=(f_1, f_2,.. f_N)$, where N is the number of features under consideration. For a given document, the feature f_j is assigned a value of 1 if the document contains the feature f_j and 0 otherwise. In the case of documents from the training set, the vector has a Label L_i for every category (class to be assigned to) as an extra component: $(f_1, f_2,.. f_N, L_1,..,L_n)$. According to the characteristics of the training set, we are interested in assigning to each new document a (classification) label. Hence, the "relevance" of the unclassified document to a predefined class is determined by the (classification) labels that are extracted from the training set. One of the main problems is related with the increasing number of features. For example if we consider every word as a feature, then the number of features is dependent on the document's length. In order to reduce the number of features, we employ algorithms that reduce the data dimensionality by using the most relevant, instead of the total set of features (words). Even though it might sound surprising at first glance, such a technique does not decrease the system's performance. Many

researchers have found that using 1-3% of the total words in a category demonstrated little or no loss in performance [LEW92][KS97][MLA98].

There are several algorithms suitable for feature selection. One such effective algorithm is the Iterative Search Margin Based Algorithm (Simba algorithm). The main reason for this choice was our interest in not only determining a solution but also in being able to measure its quality. The concept of margin is utilised in order to measure the quality of the feature set that is selected by the algorithm. A margin is a geometric measure that evaluates the confidence of a classifier with respect to its decision [SS99]. Considering that we have a set of instances assigned to a given class, the hypothesis margin calculates how much a given instance can move without changing its classification label. Figure 1 provides a representation of the hypothesis margin for an instance (colored) which can be calculated by subtracting its distances from the nearest ones that have already been assigned two different labels.

The hypothesis margin for a given instance can be calculated as $\theta_p^w(x) = \frac{1}{2}(\|x - \mu\|_w - \|x - \lambda\|_w)$ (1), where μ and λ are the nearest points to x in P with the same and different label (Fig. 1) respectively and $\|z\|_w = \sqrt{\sum_i w_i^2 z_i^2}$ (2).

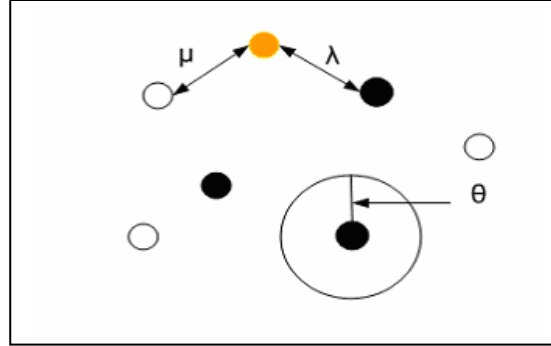


Fig 1 The hypothesis margin θ represents the distance the hypothesis can travel without assigning a different label to the instances.

The algorithm at the beginning initializes the vector w as $w = (1, 1, \dots, 1)$, and in a number of iterations T for all the instances x_i , it updates the vector w : $w = w + \Delta$, where vector Δ is calculated from the following equation:

$$\Delta_i = \sum_{x \in P} \frac{\partial \theta(x_i)}{\partial w_i} = \frac{1}{2} \sum_{x \in P} \left(\frac{(x_i - \mu)^2}{\|x - \mu\|_w} - \frac{(x_i - \lambda)^2}{\|x - \lambda\|_w} \right) \quad (3)$$

providing finally a weighted vector where N is the number of candidate features and w_j reflects the importance of feature f_j in the classification task.

2.2 Applying classifiers to the document set

Having determined the most appropriate features in a classification task, the next step is to apply classifiers. The classifier will examine the presence of specific attributes and calculate statistically their presence in respect to the ones provided in the training set. An effective classifier is the hierarchical mixtures of experts (HME). It consists of:

- Experts (functions), which model conditional probabilities, and
- Gates, which combine the probabilities of the experts.

The classification problem can be modeled using a generic function of the form: $y_i = w_i^T x$ (4), where w_i are parameters. The output of the expert network of Fig. 2 is the weighted (by the gating network outputs) result, given by: $y(x) = \sum_i g_i(x) y_i(x)$ (5), where $g_i(x)$ denotes the probability that input x is attributed in expert i .

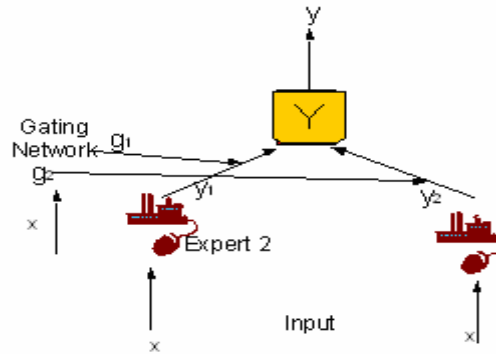


Figure 2 A mixture of experts model with two experts and two gates

The decision of whether a document can be classified as containing the described features or not, will be made based on the conditional probability $p(y|x)$ of an output y to exist, given the input x . The probability can be calculated by: $p(y|x) = \sum_i g_i(x) \phi_i(y|x)$ (6), where ϕ_i (y_i in

Fig. 2) represent the conditional densities of target Y given the expert i . In order to ensure a probabilistic interpretation for the model, the activation function g_i of the gate is chosen to be the soft-max function [BRI90]: $g_i = \exp(z_i) / \sum_j \exp(z_j)$ (7), where z_i are the gating

network outputs before thresholding. This function assures the gating network outputs achieve their sum to be equal to unity and non-negative.

Even though the ME architecture proves to be an efficient classifier, while remaining simple in its implementation, it does not behave equally well for all data-sets. This limitation can be overcome by building more complex structures, hierarchical, in which nodes of the expert model are experts themselves. Our choice to implement the Hierarchical model was driven by experiments that proved its superiority in classifying documents, even when the benchmark had specific features that make the overall task particularly difficult. It also outperforms other well-known classifiers such as the Bayesian classifier [BFG06].

Fig. 3 presents a *hierarchical* mixtures-of-experts model (HME) visualized as a tree structure. The architecture of this model consists of two levels of gates with binary branches at each non-terminal node. The outputs of the terminal experts E_3, E_4, E_5, E_6 are y_3, y_4, y_5, y_6 respectively, the outputs of the gates G_1, G_2 rooted at the non-terminal nodes in the second level are g_3, g_4, g_5, g_6 . For the outputs of the non-terminal nodes in the second we have $y_1 = g_3 y_3 + g_4 y_4, y_2 = g_5 y_5 + g_6 y_6$ and finally, the output of the system is $y = g_1 y_1 + g_2 y_2$. We have thus so far provided in our architecture an efficient way to classify unstructured knowledge sources and to manage them efficiently. We do so by organizing them in classes according to a set of predefined attributes that can be recorded when creating an appropriate training set.

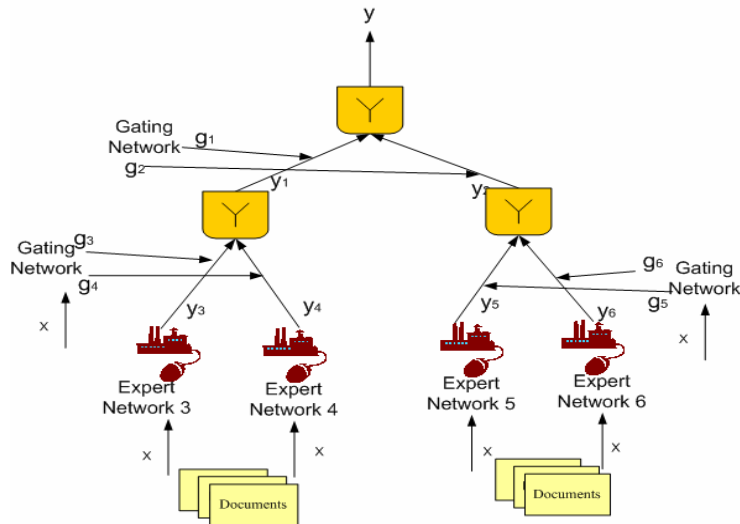


Fig 3 Two layer hierarchical mixtures of expert's architecture with a height of 2

3. Resource querying in a coalition environment

The next challenge in a coalition environment is to be able to query different domains for knowledge sources relevant to a user's query. The user can be registered in any of the domains that form the coalition. Special concern should be taken when devices with limited resources are present in the coalition, such as handheld devices. In a modern organization, knowledge assets comprise of a set of diverse nature, format and structure resources. In order to overcome semantic heterogeneity issues between the participant domains, the use of ontologies may become of vital importance [BG04]. Ontologies define the concepts of the domain of interest and their properties, and provide the basis for the semantic description and discovery of knowledge assets. Domain ontologies may be used thus in order to provide a description of the shared resources. In addition, exchanging messages between different networks is a resource consuming process (such as computational power and available bandwidth). This is an important restriction especially when computational power and energy supplies are limited (if handheld devices are used). Ontologies can be exploited towards this direction by providing the ability to classify an entire domain according to thematic areas.

Efficient querying involves identifying whether a specific domain contains knowledge sources of specific interest to a user query. Thus two phases are necessary: first a classification of documents according to pre-specified categories. Second, following the user query, the domain ontology rapidly provides information so that the query may be considered as irrelevant to the domain's subject areas. The different ontologies from the participant domains are gathered in a global repository [GRI06]. The user query is primarily directed to the Global Ontology Repository, which stores all the domain specific ontologies. Each domain-ontology independently is checked, in order to identify which domains contain knowledge sources relevant to the imposed by the user query; for all the domains not containing relevant information, the query is excluded from further consideration. If the domain's knowledge sources are relevant, the query is further forwarded to the domain's nodes. Thus, we have achieved to categorise the different domains according to a broader set of topics, encoded in appropriate ontologies and constituting each a Virtual Overlay Network (VON) [MPB05a] [MPB05b] [GBK06]. Fig. 4 represents the concept of VONs. In a use case scenario, the system first classifies the documents to predefined classes, and based on specific features present in the documents it can form appropriate ontologies; these ontologies can help in estimating the relevancy of a user's query to the knowledge categories that each domain's assets belong to.

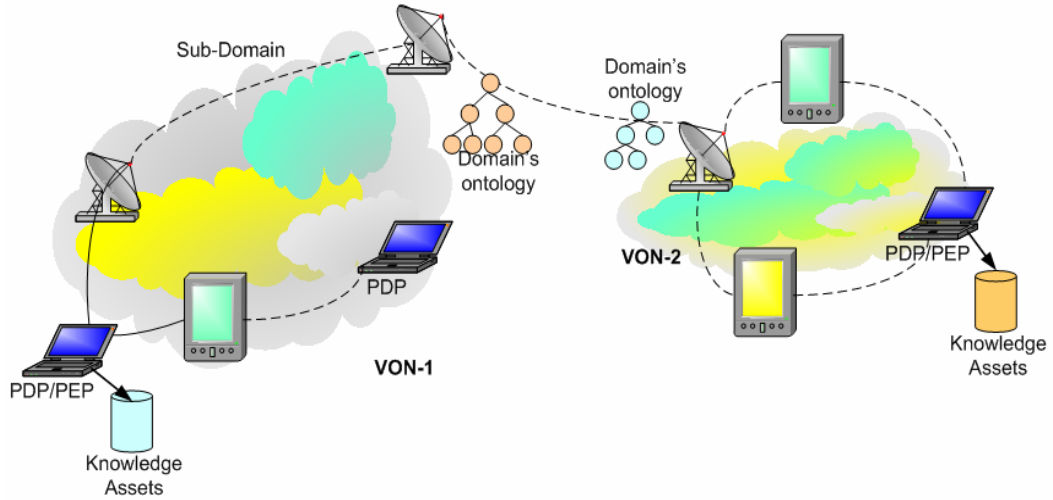


Fig 4. Independent domains classified in different Virtual Ontology Networks (VON's) according to the ontologies of each domain.

4. Access Control framework

4.1 Constraint specification and Access Controls

The next step upon identifying knowledge assets relevant to a user's interests is to establish an access control framework suitable for multiple policy environments. In such an environment, there is a necessity to establish a mechanism allowing interoperation between the participant domains. Even though single domain security models have been adequately researched, there is a lot of ongoing work in the area of federated environments. In the next paragraphs we describe our design choices. We firstly present a scalable solution based on the notion of policy mappings; the proposed framework is further optimized, by introducing an appropriate formalism and also by introducing a technique seeking optimal weighted paths between the role hierarchies.

We consider that all the domains conform to the principles of the Role Based Access Control (RBAC) model. A complete RBAC model [SAN00] includes the following variables and functions:

- The sets U (users), R (roles), P (permissions) and S (sessions)
- User to role assignment $UA \subseteq U \times R : U \rightarrow 2^R$
- Permission to role assignment $PA \subseteq P \times R : R \rightarrow 2^P$
- A mapping of sessions to a single user assignment $US : S \rightarrow U$
- A mapping of sessions to the set of roles associated with each session $S \rightarrow 2^R$
- A partial ordering $RH \subseteq R \times R$, represented by the symbol: \geq , which defines role hierarchy. $R_1 \geq R_2$ implies that R_1 inherits permissions from R_2 .

We can therefore consider as $U = \{U_1, U_2, \dots, U_n\}$ the set of users, which map to a set $R = \{R_1, R_2, \dots, R_m\}$ of roles, and we can also consider as $O = \{O_1, O_2, \dots, O_3\}$ a set of shared resources. Additionally access attributes may be considered members in a totally ordered set $A = \{0, w, x, r, wx, wr, wx, wrx\}$ (combination of values w, r, x as denoted in UNIX notation). In RBAC access control restrictions may be encoded as a triplet of the form $\langle R, O, A \rangle$. The aforementioned role and permission structures may then be considered as members in an ordered hierarchy, which can be graphically represented in Fig. 5

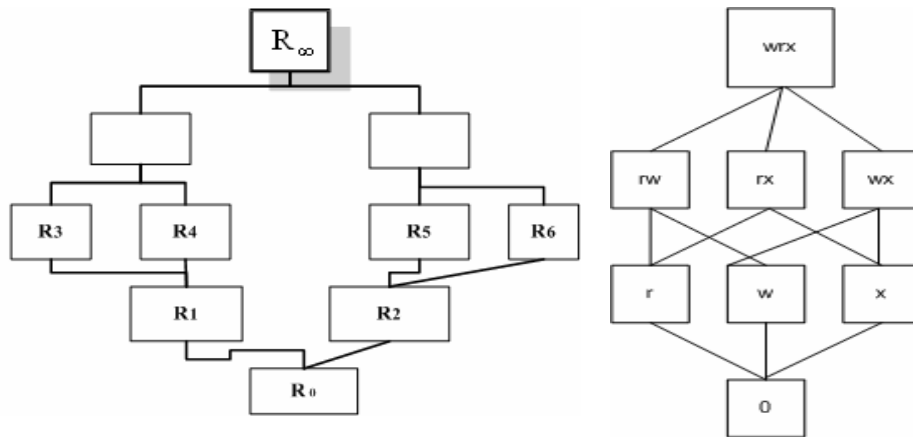


Fig 5 Permission and role ordered hierarchy representation

In multi-domain environments we are interested in assigning privileges to users belonging to another domain. If we attempt to classify permissions independently for each user for both local and remote domains, by maintaining separate access matrices, the system's complexity will raise significantly. We adopt instead, the solution of policy mappings [MPB05a][BGK05][MPB05b] that allow the determination of corresponding roles from one domain to another. Mappings are stored in a coalition repository, which acts as in the case of [MAW05] [MPB05b].

In order to reduce the information load that is stored in the coalition registry, we have introduced a flexible solution that allows roles from different domains to be mapped onto a single generic role. We propose constructing a generic role hierarchy that acts as mediator ontology for the mapping process [BGK05]. All the mappings between the participant domains direct from the local domains to the generic ontology and vice versa. Thus, the coalition repository contains less information, consisting only of policy mappings between the central ontology and the destination authorization hierarchies. The mappings are pre-settled by the coalition administrators, who are aware of the needs of each role under the federated environment. The global role-schema hierarchy assumption to which local policies map is not restrictive; typically though it is usual that many agencies or ministries have similar (but not identical) roles. For example ministries have ministers, general secretaries, department directors (or sector managers) and so on. Even though the roles in departments may differ in number or name from one domain to another, in a coalition environment we could have roles from different departments performing similar tasks.

More formally we may consider a role mapping as a function that maps a role of domain A to a role of domain B. $M_{AB}:R_A \rightarrow R_B$. By virtue of this role mapping, any user that has a role R_A in domain A, is allowed to perform all the actions for a role R_B in domain B [SJB05]. The role mapping assumption need not necessarily be two-directional. Thus, when a role A with equal privileges to the ones given to role B of another domain, it is not necessary that role B will also get role A's privileges. We distinguish two types of mappings, in-mappings towards the central ontology, and out mappings directing towards local roles.

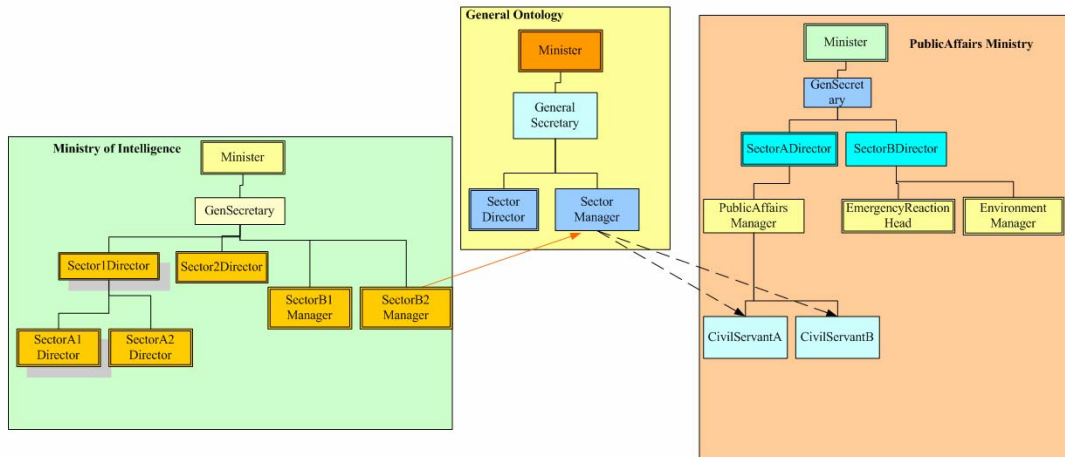


Figure 6 Role mappings across different domains. A local role on domain A maps to the global role hierarchy and accordingly through it, to two different roles at domain B. In our case the domains are two different ministries.

4.2 Role - Representation issues

For interoperability reasons, the role relevant information needs to be encoded in a suitable language. Policy languages are suitable for this purpose. They enable encoding of the access control information in a machine and humanly interpretable manner. It is for these reasons that we have selected the XACML [XACML] policy language.

Among XACML's strong points, are:

- It is standardized and it is open, allowing extensions that enable interoperation between various platforms
- It is codified in (XML) which tends to dominate as codification standard and is operating system independent.
- It allows context based authorization, which is a big advantage

For policy editing instead of using XML we have selected a more expressive framework, the RDF [BEC02]. By doing so, we can incorporate in our policy documents context related information, such as domain-related information. Table 1 presents a fragment from a policy specification document in RDF. The <activation-time> and <deactivation-time> tags provide information about the time intervals within which a role can be active, while the <supervises> tag enables representation of role hierarchies.

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF>xmlns:rdf="http://www.w3.org/1999/02/22-rdf-s
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:base="http://defenseMinistry.gov/roles"
xmlns:prm="http://defenseMinistry.gov/permissions">
<rdf:Description rdf:ID="GenSecretary">
<prm:activation-time>9:00</prm:activation-time>
<prm:deactivation-time>23:00</prm:deactivation-time>
<prm:DomainDescription>intelligence.defense.org</prm:D
<prm:supervises parseType="Collection">
<rdf:Description rdf:ID="Sector1Director"/>
</prm:supervises>
</rdf:Description>
<rdf:Description rdf:ID="Sector1Director">
<prm:activation-time>9:00</prm:activation-time>
<prm:deactivation-time>17:00</prm:deactivation-time>
<prm:DomainDescription>intelligence.defense.org</prm:D
<prm:supervises parseType="Collection">
<rdf:Description rdf:ID="SectorA1Director"/>
<rdf:Description rdf:ID="SectorA2Director"/>
</prm:supervises>
</rdf:Description>
</rdf:Description>
</rdf:Description>
</rdf:RDF>

```

Table 1. RDF-based Role attribute and hierarchy definition (fragment)

Similar to the policy which is edited in an XML based platform, policy mappings also need to be edited in an interoperable platform. We have chosen the XPATH language, defining in the XML tree, both in-mappings and out-mappings. XPATH aims in addressing parts of XML documents. It represents location of data in an XML document correctly and efficiently, which makes it a suitable language for both XML query and access control. An example mapping based on XPATH is presented in Table 2, where roles from one domain are mapped to another domain’s roles indirectly through the central role hierarchy.

Ministry of Intelligence	CENTRAL
Minister/GenSecretaryB/SectorB2Manager	Minister/GeneralSecretary/SectorManager

Table 2a. XPATH based role mapping (in-mapping) between local and central ontology hierarchy roles

PublicAffairsMinistry	CENTRAL
Minister/GeneralSecretary/SectorADirector /PublicAffairsManager/CivilServantA	Minister/GeneralSecretary/SectorManager
Minister/GeneralSecretary/SectorADirector /PublicAffairsManager/CivilServantA	Minister/GeneralSecretary/SectorManager

Table 2b. A role from the Central hierarchy maps to two local roles (out-mapping)

Therefore we define paths that allow the mapping of roles between different role schemata. Notice that due to the expressiveness of XPATH, one can represent more complex role mappings in a very compact way, by grouping together equivalent roles in one XPATH expression. In this way one need not write separate rules for each role. Moreover, we have to note that mappings can be easily codified in the specific purpose coalition registry, enabling to form coalitions in a robust way, and in short time which is an important requirement in coalition formation.

In our approach, we also consider that mutual trust exists between the different domains that participate in the coalition. Other security models for distributed environments utilise the concept of trust for authentication and access control enforcement, by assigning privileges to specific credentials associated with a user. In trust-based models, it is possible to assign privileges even to unknown users that first time interact with the system, something that makes them suitable for environments with absence of well defined policies (such as eCommerce environments or the Internet). Our approach though, considers autonomous

domains which interoperate under a common framework that rules the coalition. In our framework all the roles are well defined and there is no risk such as these introduced when unknown users interact with trust based systems. In our model, following the approach off [AM03], we consider that interoperation between different policies is substitute to a number of bilateral arrangements under a supervising mechanism or other regulating framework (such as a governmental framework) This restriction is necessary, since the problem of merging different policies in the absence of mutual trust is NP-complete [BB03].

4.3 Access Control Architecture

Our access control architecture builds upon the main XACML operational principles. Its main modules are 3, namely the Policy Enforcement Point, the Policy Decision Point and the Context Manager (Fig. 7). The Policy Enforcement Point (PEP) responsible for applying the policy decisions, while the Policy Decision Point (PDP) evaluates requests by examining the user's credentials and interprets the request against the domain's policy. The Context Manager (CM) holds the responsibility for collecting and sending context related attributes to the PDP (i.e. domain specific information). We have also integrated within the PDP the coalition registry that stores information about the incoming mappings for a domain. We have chosen a centralised implementation for the PDP, but in order to avoid a single point of failure or potential bottlenecks it can be deployed in a distributed manner as described in [MPB05b], [MAW05].

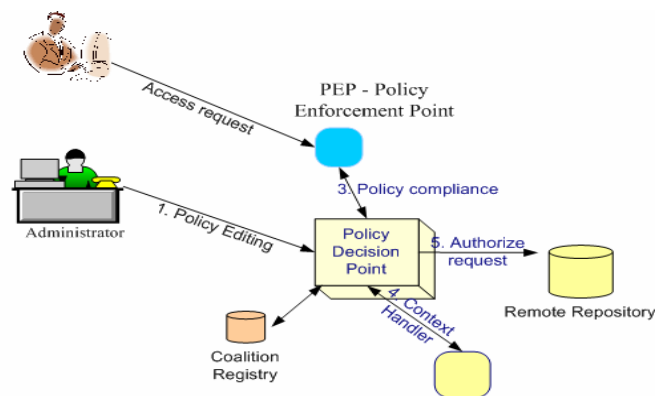


Fig 7 (Local and remote) Authorization process in steps, using XACML basic operational principles

The overall operation of this multi-domain authorization framework can be described in the following (Fig 7). The policy administrator is responsible for editing the policy and making it available for the domain, through the Policy Decision Point (PDP). When a request for a resource appears it has to be validated for its consistency with the local security policy prior to its execution. In case of a request from a remote domain, the available mappings are first retrieved from the coalition repository. Accordingly, each request (from the same or from a remote domain) is directed to the Policy Enforcement Point (PEP). The request is constructed in an appropriate XML message and directed to the PDP. Prior to the validation of the request, the Context Manager (CM) is sending additional subject, resource, action and environment attributes to the PDP. Accordingly, the request is validated from the PDP and a response message is sent back to the PEP, which handles the details about providing authorization to the requester. Each domain maintains its own PDP and PEP.

This access control architecture, that implements the solution of policy mappings, is responsible for the access control enforcement as part of the overall system. Typically, the authorization module is the third in sequence utilised module for each user query. The classification module is responsible for distinguishing the knowledge sources, the Virtual Ontology Networks assist in identifying the most relevant sources in each domain, and lastly,

the authorization module checks for a remote user if there exists an available mapping so that sufficient privileges can be assigned to him/her.

5. Formalizing coalition based access controls using soft constraints

Security management of coalitions is subject to a large number of restrictions and continuous changes. Policy mappings are a simple, yet effective solution. There is though a necessity for a form of optimisation so that the engagement of administrators is not constantly necessary. In the following paragraphs we will explain a technique that allows such an optimisation; we will utilise for our framework a formalism that allows modelling both the single-domain and multi-domain access control. Our main source of inspiration is lattice theory [KABS06] and the algebraic structure of semirings which we apply to solve constraint problems, adopting the notation appearing in [BIS04][BMR04]. We will prove that this formalism is efficient to describe the role and permissions hierarchy and lastly, we will use the same formalism to show how our proposed policy mappings can be optimised using this framework.

5.1 Setting the scene - Semirings and Soft constraint satisfaction problems (SCSP's)

Semirings are useful algebraic structures, capable of modelling a variety of problems. In [BB03] [BGK06] [BIS04] they have been applied towards security oriented problems. We will begin with a brief introduction on semirings and soft constraint problems in general and we will show how they can be applied in security oriented problems.

Definition 1: A semiring S is a tuple $\langle A, +, *, \mathbf{0}, \mathbf{1} \rangle$ where

- A is a set with $\mathbf{0}, \mathbf{1} \in A$
- $+$ the additive operation is closed, commutative and associative over A with $\mathbf{0}$ as the absorbing element
- $*$, the multiplicative operation is closed and associative over A with $\mathbf{1}$ as its identity element and $\mathbf{0}$ as its absorbing element
- $*$ distributes over $+$

We have to note here the fact that the $+$ operation is defined over sets and not over pairs or tuples. This constitutes it as commutative, associative and idempotent.

Definition 2: A constraint system is defined as a tuple $CS = \langle S, D, V \rangle$ where S is a c-semiring, D is a finite set, and V is an ordered set of variables.

Definition 3: A constraint over such a system is a tuple $\langle \text{def}, \text{con} \rangle$ where $\text{con} \subseteq V$ is the connection function $\text{con}(c) = \langle v_1, \dots, v_k \rangle$ that describes which variables are involved in which constraint, while def is the definition function and specifies which are the domain tuples permitted by the constraint. Thus def assigns a value from the semiring to each combination of values of the variables in con . This value can be a probability, a cost, a preference etc.

We can use the aforementioned structures to formulate our access control problems using appropriate semirings to model permission and role hierarchies. Thus, the inclusion relation enables us to model the ordered role and permissions hierarchy. Having appropriately defined the two operations for role and permissions hierarchies we can cast the problem of user to role and permissions assignment to a CSP problem. The advantage of using c-semirings is that the idempotency of the $+$ operation helps to define a partial ordering \leq_s over the set A , which enables comparison of different elements of the semiring. Typically $a \leq_s b$ is equivalent to $a + b = b$, meaning that one element is preferable between a, b over the operation $+$. Thus we can model actions where a role or permission will be preferable depending on whether we want to assign the least permissions or the opposite. Two operations are of particular interest when dealing with multiple constraints: the combination and projection.

Definition 4 (projection): Given a constraint system $CS = \langle S, D, V \rangle$ where $S = \langle A, +, \times, \mathbf{0}, \mathbf{1} \rangle$ a constraint $c = \langle \text{def}, \text{con} \rangle$ over a system as the aforementioned one, and a set of variables $I \subseteq V$, then the projection of c over I written as $c \downarrow_I$, is a constraint $\langle \text{def}', \text{con}' \rangle$ with $\text{con}' = I \cap \text{con}$

$$\text{and } \text{def}'(t') = \sum_{\{t \mid t \downarrow_{I \cap \text{con}}^{\text{con}} = t'\}} \text{def}(t)$$

Typically, projection means elimination of domain values over the variables of interest.

Definition 5 (combination): Given two constraints c_1, c_2 over the proposed system, their combination $c_1 \otimes c_2$ is another constraint $c = \langle \text{def}, \text{con} \rangle$ where $\text{con} = \text{con}_1 \cup \text{con}_2$ and $\text{def}(t) = \text{def}_1(t \downarrow_{\text{con}_1}^{\text{con}}) \times \text{def}_2(t \downarrow_{\text{con}_2}^{\text{con}})$. It is obvious that \otimes is both commutative and associative, since the \times operation is.

The solution $\text{sol}(P)$ of a constraint problem $P = \langle C, \text{con} \rangle$ over a constraint system CS is defined as $\text{sol}(P) = (\otimes C) \downarrow_{\text{con}}$, which means that we have to combine all the constraints under consideration and then project them over the variables defined by function con .

The optimum level of consistency $\text{oLevel}(P)$, is a useful metric that yields an estimation of how much the solution satisfies the constraints of the problem and is obtained if we first calculate the solution and then project it over the empty set of variables.

5.2 Defining appropriate semirings

Defining an appropriate semiring is an interesting problem, since it affects significantly the identification of an appropriate solution (if there is one). In our problem we can identify two key-hierarchies in RBAC, more specifically the role hierarchy and the permissions hierarchy.

The role hierarchy can be represented by the role semiring: $\langle R, +_R, *_R, R_0, R_\infty \rangle$, where

- R is the set of roles in the system
- The $+_R$ operation is defined as: $(R_1 +_R R_2)$ is the highest common descendant of roles R_1 and R_2 in role hierarchy
- The $*_R$ operation is defined as the common ancestor of roles R_1 and R_2 in role hierarchy
- R_∞, R_0 are the roles with maximum and minimum privileges. For example in the hierarchy of Fig 1a the role R_∞ has maximum privileges, while R_0 implies the role with minimum privileges.

The utility of the above constraint system can be proved as follows [BB03]: considering that a system offers authorization to remote users, we can pick the constraint $\langle R, V, C \rangle$ where R is the roles semiring, C is the set of credential types known to the system and V is the set of values these credential types can take. So under these circumstances, each existing tuple equals to the assignment of a role R to this combination of values and role R_0 to every other tuple.

Next, we consider the permission hierarchy and we define the appropriate semiring $\langle P, +_P, *_P, P_\infty, P_0 \rangle$, where

- P is the set of permissions in the system
- The $+_P$ operation is defined as: $(P_1 +_P P_2)$ is the highest permission between P_1 and P_2
- The $*_P$ is defined as the lowest permission
- P_∞, P_0 are highest and lowest permission in the hierarchy respectively.

The permissions semiring can be utilised as follows [BB03]: We consider the constraint system $\langle R, O, P \rangle$ that consists of the possible values for variable R (roles), O (the objects to

be accessed) and P (the necessary permissions). Every role in this system is assigned a tuple t of access rights. The result is a SCSP a solution of which is, in every case, the lowest role in the hierarchy that is required to access the resource under consideration.

In order to achieve a solution to the problem of assigning roles and defining access rights over the coalition workspace, we perform the following actions:

We need to define a new composite semiring system, $\langle P, L_1+L_2, R \rangle$ where P is the appropriate semiring, L_1 and L_2 are the sets of domain local roles and R is the set of global roles. Then, in order to establish a solution to the problem of assigning permissions to remote roles we have to work as follows: we first built a CSP by assigning permissions to local roles $P \rightarrow L_1, P \rightarrow L_2$ and then we make the assignment of local roles to global roles; thus we get the permissions assigned to the global roles (the central hierarchy roles). Now in case the CSP describing the permissions assigned indirectly to a global role dominates the CSP of the target role we have an acceptable solution, meaning that we have appropriate permissions to access the target role's shared resources. Policy mappings can be considered as some tuples that are known from the beginning. The CSP may be built by retrieving the necessary information from the coalition registry; depending on whether the CSP has a solution or not, the request either is satisfied or (in case it is not) the user is not granted access to the specific resources.

5.3 Optimizing policy mappings – releasing information in a controlled environment

In many cases a mapping may not be present; still it may be allowed for a remote role to access the requested resource. In this case the engagement of the administrators would be necessary in order to adjust the system to grant access to a specific resource. Our concern is to consider the cases where this overhead for the administrators could be avoided. There are situations that information flow may be allowed from senior roles to junior non-critical roles. In this case, we can optimise the system's performance. The functionality of such an approach is similar to that established by release control policies as described in [YWJ05]. In these approaches there is a filtering enforcement point that in our case can be incorporated within the PDP. The main principle behind this idea is that implicit authorizations can be derived from explicitly declared authorizations in the usual manner by propagating them downwards the authorization hierarchies. Such propagations enable higher entities to generalise lower entities [YWJ05].

In such a case given a triplet (Object, Sender, Receiver) we would like to determine a legitimate path that allows information flow in a secure manner. For clarification, we should not allow some role with fewer privileges than the required ones to access any resources. By using the SCLP formalism, we can cast this problem to a multiple criteria weighted shortest path one.

Consider the following scenario (Fig. 8): role V that holds a high position in the authorization hierarchy of domain B, wants to access resources from domain A allowed to be accessed by role T2 or his superior roles (as the role inheritance assumption stands in the standard RBAC model). There is no established mapping to grant him permissions from role T2; therefore, the administrator's engagement would be necessary. Given that role V may acquire permissions from role U in domain B and that there is a mapping towards role Q in domain A which may inherit permissions from role T2, there is a way to consider that role V could ordinarily be assigned permissions from role T2. The problem then can be cast to identifying such legitimate paths (if they exist). Of course, there are cases where we would not like to activate a role because of its criticality. We assume also that this technique does not necessarily apply in situations where data may be considered as sensitive (still even in that case the policy mappings work effectively); in such a case we could use the optimization techniques as a support tool to provide suggestions, while it will be the administrator's responsibility to verify and possibly activate the proposed solution.

In order to consider cases where some roles are more critical and thus less desirable to be activated, we do not simply seek for a path's existence, but we calculate paths with weights. A weight may consist of two values assigned to each node $\langle a, b \rangle$ where "a" represents the height of a role in the hierarchy and "b" the criticality of a role. By using a path evaluation algorithm we can identify all legitimate paths. When a user seeks a legitimate release path, from a given sender to a given receiver, the user does not want a set of paths, but rather an optimal path instead, based on certain assigned weights. According to access control principles, we have considered two main restrictions: i) a user cannot obtain the permissions associated with a user higher in the hierarchy and ii) roles that are more important (critical) are less likely to be activated.

5.4 Identifying optimal paths

Consider the case of Fig 8 with the two role hierarchies. We can represent the roles in this hierarchy by considering a graph $G=(N, E)$ where the roles are represented as nodes in the graph and assign a weight to each arc $e \in E$ from node p to node q ($p, q \in N$). This weight will be the aforementioned pair of values, associated with the level of each role in the hierarchy (a parameter that defines how important a role is in the organizational hierarchy) and the criticality associated with each role. Now this example may be modelled by two semirings. For the first parameter, we can define a semiring $\langle \mathbb{N}, +, \min^*, 0, +\infty \rangle$ where \min^* defines the minimum difference (considering only positive differences) and the $+$ operation with the classical meaning. For the second parameter of the weight, related with the criticality, we define a semiring $\langle \mathbb{N}, +, \min, 0, +\infty \rangle$, where \min and $+$ are defined with the classical notion. Now according to the principles described in the previous paragraphs, we can identify legitimate paths. We do not wish to violate the role hierarchy principles by moving to roles with more privileges and we wish to avoid activating roles of critical importance. For this purpose, we can use well known algorithms for shortest paths. Nevertheless, calculation of path weights is a complicated process.

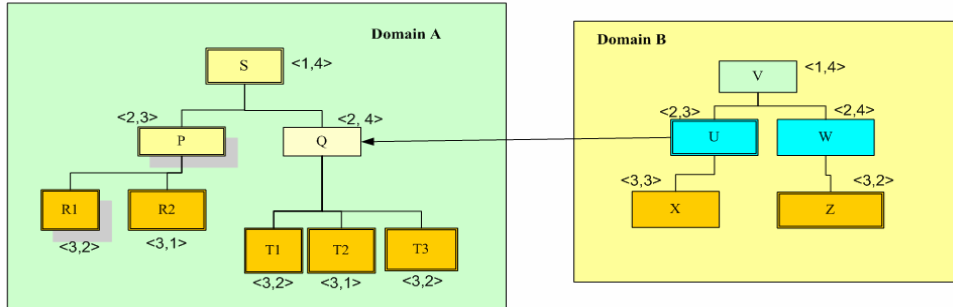


Fig. 8 Example of a role mapping and role hierarchy representation with costs

Our problem can be formulated as a Soft Constraint Logic Programming (SCLP) [BIS04] [BGK06] problem, which works over an appropriate semiring. In order to find a path that does not violate hierarchy constraints, we calculate the differences between the first parameters in the weights that are related with each role's position in the hierarchy. We only allow positive differences (or equal to zero), meaning that the target role has to be lower in the hierarchy (we consider that different hierarchies and positions at the same depth are equivalent). Additionally we want to calculate the differences of the second weight values, so that the criticality of the assigned path is minimal.

We will proceed with the example of the previous section, represented in Fig. 8, where role V from domain B wants to access resources assigned to role T2 from domain A. There is a direct mapping from role u to role q.

The calculation to find the total cost for the path from V to U will work as follows: $[c_{vu}: \langle (2-1), (4+3) \rangle = \langle 1, 7 \rangle]$. The first instance of c_{vu} is calculated by subtracting the hierarchy

differences (considering they are positive) which are calculated by the term $\sum_i \sum_{j,i \leq j}^{i,j:neighbours} (x_i - x_j)$, while the second instance given by the term $\{\min[\sum_i \sum_j^{i,j:neighbours} (y_i + y_j)]\}$ counts the sum of criticalities (which can be set arbitrarily), aims to hinder administrators from activating these intermediate roles unnecessarily. Accordingly, for transition from u to q we have, $[c_{uq} : \langle 2-2, 4+3 \rangle = \langle 0, 7 \rangle]$. At last, we have from Q to T2: $[c_{QT2} = \langle 3-2, 1+4 \rangle = \langle 1, 5 \rangle]$. In this case, we have identified (the only one) legitimate path. In case we had multiple mappings and multiple paths, we would choose the one that minimises the criticalities sum.

We could alternatively model the system by monitoring the behaviour of other parameters instead of the criticality. Also, we could use the proposed technique as an administrator's support tool and it is not necessary to allow role assignments, unless an explicit role mapping exists. We could remark that by modelling the network with the proposed approach, we enable policy merging to a high extent, retaining hierarchy-related restrictions and thus enabling a secure and scalable solution for the problem of secure interoperation.

5.5 Overall System architecture

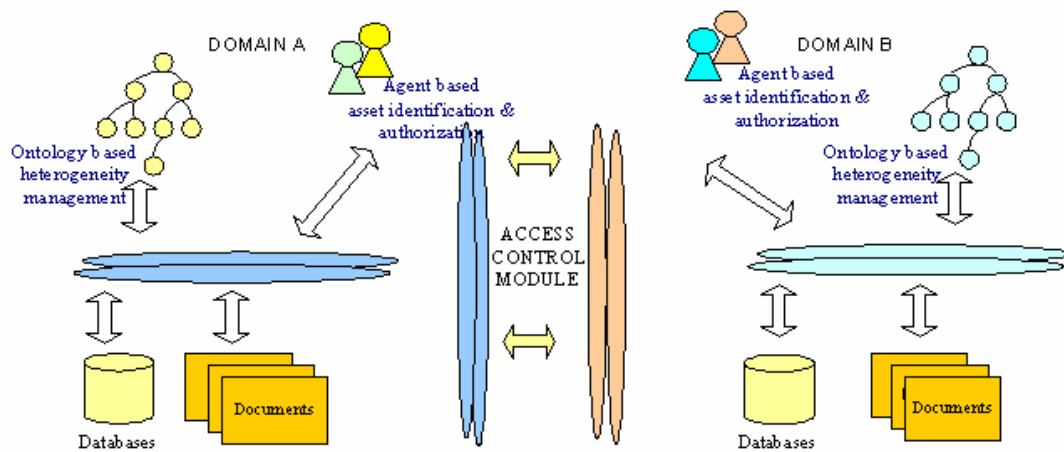


Fig 9 Overview of the overall system architecture

We have proceeded in building a prototype implementation, which consists of different modules:

- The document classification and management module, which classifies documents to different classes according to the training instances provided by the users; it also codifies knowledge in a structured manner (database).
- The access control framework, responsible for the enforcement of security policies. The access control framework consists of the PDP, the PEP, the Context Manager for each domain, and the coalition registry. The queries and responses are encapsulated in agent exchange messages; typically one agent per domain may be assigned to carry the user credentials and provide them to the PDP which will reason accordingly for the legitimacy of the request. Thus the process remains transparent to the user, who simply has to provide a username/password combination. The application is then responsible to provide him/her authorization to access resources from different domains.

- The ontology management module which enables to classify each domain's assets to different subject areas. In this way we are firstly able to overcome heterogeneity issues and secondly we facilitate interoperability by defining the key-terms in the agent communication vocabulary. The task to automate the asset query and retrieval process is assigned to a domain, search agent.

For each domain, a pair of agents is assigned, one agent responsible for querying different domains for knowledge resources relevant to a user's query, and one responsible to authorize the request, by providing the user's credentials to the PDP. Thus the S-Agent (search agent) is querying firstly the domain's ontology to identify the relevancy of the request with the contents of the domain; the A-Agent (authorization agent) is providing the user credentials to the PDP. The agent's purpose is to make the process transparent to the user, which does not have to query or request authorization for each domain separately; instead, these processes are automated and transparent.

Our solution is characterized by:

- Robustness, since we have provided a simple, yet effective solution to assign permissions for users originating from different domains; in addition, the principle of security and autonomy as defined in the introduction of this paper are retained for all participant domains.
- Scalability, because the way mappings are defined enables the system to grow without imposing additional costs and without raising the system's complexity management.
- Flexibility; there is no need for continuous monitoring and adjustment of the system, since we have provided a way to optimize the system's performance.

6. Related Work

6.1 Distributed Knowledge Management Architectures

Many distributed knowledge sharing infrastructures have appeared lately, mainly utilizing peer-to-peer technologies. Our work on the contrary, enables different domains to cooperate, while allows them to retain their autonomy.

Edutella [NWO02] is a peer to peer system that utilizes RDF ontology to manage metadata. It is mainly designed to facilitate knowledge sharing between different participant domains, such as academic environments. Edutella has the ability to control information flow in order to avoid bottlenecks. Its security model is based on the idea that different nodes loan out their credentials building communities of trust. In our model instead, we propose an RBAC oriented solution, where different autonomous systems merge their policies while adhering to the main RBAC principles.

XAROP [TEM04] is a peer-to-peer system, which manages heterogeneous knowledge sources by using ontologies. Determination of access privileges is performed by a manual assignment of privileges to groups of users, defined within the XAROP framework. Thus its scalability potential can be limited, while determination of access privileges is not defined in a flexible manner.

ADAM [SEL04] is a distributed system, which utilizes trust based negotiation procedures for the establishment of transactions between users. Its architecture is agent based, with one agent being responsible for gathering the knowledge from distributed nodes and a second one for handling the authorization processes on behalf of the user. ADAM is mainly utilizing the trust model, which is suitable for environments without well defined organizational policy. In ADAM users establish contacts based on the recommendations they acquire about someone's

reputation. This model is mostly suited for open environments, while our model on the other hand focuses on merging different autonomous systems and maps their policies in a flexible and scalable manner.

6.2 Access Control Models for coalitions

The problem of defining access control models for multi-domain environments has recently attracted considerable interest. A number of solutions have been proposed towards this direction.

In [SJB05] a policy merging algorithm is defined allowing the determination of a global policy, based on a merging process of the individual access control policies. Following that, a conflict resolution process is performed that attempts to remove conflicting permission to role assignments; the disadvantage of this method is that it is hard to reflect policy updates since the policy merging algorithm requires polynomial time. In our work, policy updates are easily integrated in the policy interpretation mechanism and at the coalition registry, while there is support to define additional optimal paths avoiding the administrative overhead.

In [BB03] a framework is proposed that builds upon SCLP's for multi-domain cooperation. In this framework appropriate semirings are defined that are capable of assigning permissions to local roles and accordingly permissions over the shared workspace. The intersection of the two semirings enables to define whether the shared workspace is achieved or not. Our work provides a more generic solution while it extends this framework, by also defining policy mappings (these can be considered as tuples that are known to satisfy the SCSP and help to identify always a solution to the problem of coalition access control). This solution in our case, in comparison to [BB03] works even in case of sensitive environments. In our work we also introduce a form of optimization by introducing the determination of safe release paths in a similar manner to release control policies mentioned in [YWJ05].

Khurana et al. [KGL02] define a model for the dynamic management of coalitions based on the RCL 2000 RBAC-oriented language. Coalition formation is performed as a round-robin negotiation where domains make proposals about the management of shared coalition assets resources. The main idea of this approach is that domains make proposals about shared resources, then a set of global users is formulated, a coalition access matrix keeps records of global roles and permissions are assigned to them. Even though this solution is pretty flexible and facilitates automated negotiation, it is difficult to scale.

In [MAW05] [MPB05b] two different scalable solutions supporting the dynamic formation of coalitions are proposed. They mainly utilize a distributed service registry, similar to the coalition registry introduced in our approach. In our work though, we provide a formal framework to support the formation of coalitions while we also introduce an optimization technique to further optimize the overall system's performance.

7. Conclusions

We have provided throughout this paper an access control framework for dynamic coalitions. In order to allow for secure information flow in dynamic coalitions, the concept of policy mappings has been introduced that allows assignment of permissions to users originating from a collaborating, remote domain. We have formulated the problem of access control for coalitions as a SCSP, and we have provided a form of optimization by allowing information release through legitimate role paths.

The validity of the proposed approach has been proved by means of a proof of concept implementation; its main modules have been analyzed throughout this paper: the document classification module, that utilizes two effective algorithms for efficient classification of documents; the ontology based module, that facilitates querying of the network; lastly, the

access control module that builds upon our security framework, while for its deployment we used standardized policy languages. The main difficulty in our approach is to build the generic ontology that acts as the federal one. In most cases though, for organizations that work under the same framework (ministries, hospitals) it is relatively easy to define a generic ontology. In cases where the structure of the participant organizations does not allow this, policy mappings can still be applied without the intermediate ontology.

We have also described our prototype implementation, parts of which were highlighted also in [BGM05][GRI06][MPB05b][GBK06] that integrates the ability to manage knowledge assets effectively under the distributed cooperation environment, while it also implements an access control mechanism for dynamic coalitions.

We are currently working towards expanding the capabilities of our framework by incorporating in the negotiation phase domain preferences over resources and roles; such preferences are considered during the merging process and the degree of satisfaction can be measured through fuzzy preferences.

8. References

[AM03] Ao X. and Minsky N. H., Flexible regulation of distributed coalitions. In LNCS 2808: the Proc. of the European Symposium on Research in Computer Security (ESORICS) 2003

[BB03] Bharadwaj V., Baras J, “Towards automated negotiation of access control policies”, in *Proceedings of 3rd IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY'03)*, 2003

[BEC 02] Beckett D., ed., RDF/XML Syntax Specification, W3C Recommendation, www.w3.org/TR/rdf-syntax-grammar.

[BFG06] P. Belsis, K. Fragos, S. Gritzalis, C. Skourlas, SF-HME system: A Hierarchical Mixtures-of-Experts classification system for Spam Filtering, *Proceedings of the Proceedings of the 21st ACM Symposium on Applied Computing ACM SAC 2006 – Computer Security Track*, G. Bella, P. Ryan (Eds.) (Eds.), pp. 354-360, April 2006, Dijon, France, ACM Press

[BG04] P. Belsis, S. Gritzalis, Distributed Autonomous Knowledge Acquisition and Dissemination Ontology based Framework, *Proceedings of the PAKM 2004 5th International Conference on Practical Aspects of Knowledge Management -Workshop on Enterprise Modeling and Ontology: Ingredients for Interoperability*, H. Kuhn (Ed.), pp. 100-104, December 2004, Vienna, Austria, University of Vienna

[BGK05] P. Belsis, S. Gritzalis, S.K.Katsikas, A Scalable Security Architecture enabling Coalition Formation between Autonomous Domains, *Proceedings of the Proceedings of the 5th IEEE International Symposium on Signal Processing and Information Technology (ISSPIT'05)*, pp. 560-565, December 2005, Athens, Greece, IEEE Press

[BGK06] P. Belsis , S. Gritzalis, S.K.Katsikas, Optimized Multi-Domain Secure Interoperation using Soft Constraints, *Proceedings of the Proceedings of the 3rd IFIP Conference on Artificial Intelligence Applications and Innovations (AIAI 2006)*, M. Bramer, I. Maglogiannis (Eds.), pp. 78-85, June 2006, Athens, Greece, Springer

[BGM04] Belsis P., Gritzalis S., Malatras A., Skourlas C., Chalaris I, “Enhancing Knowledge Management through the use of GIS and multimedia” in *Proceedings of Practical Aspects of Knowledge Management (PAKM 2004)*, Vienna Austria, LNAI vol. 3336 Springer, pp. 319-329, 2004

[BGM05] Belsis P., Grizalis S., Malatras A., Skourlas C., Chalaris I., “Sec-Shield: Security Preserved Distributed Knowledge Management between Autonomous Domains”, in

Proceedings of the 2nd International Conference on Trust and Privacy in Digital Business (Trust Bus 05), Copenhagen, Denmark, LNCS Springer, 2005

[BGS05] P. Belsis, S. Gritzalis, C. Skourlas, Security Enhanced Distributed Knowledge Management Architecture, *Proceedings of the Proceedings of the 5th International Conference on Knowledge Management*, K. Tochtermann, H. Maurer (Eds.), pp. 327-335, July 2005, Graz, Austria, JUCS Pubs.

[BIS 04] Bistarelli S., "Semirings for Soft Constraint Solving and Programming", Springer Lecture Notes in Computer Science, Vol. 2962, 2004.

[BMR04] Bistarelli S., Montanari U., Rossi F. "Semiring-Based Constraint Logic Programming: Syntax and Semantics, *ACM Transactions of Programming. Languages and Systems (TOPLAS)*, ACM Press, Pages: 1–29, Vol. 23, issue 1, 2001

[BON02] Bonifacio M., Bouquet P., and Traverso P., "Enabling distributed knowledge management. Managerial and technological implications", *Informatik – Informatique*, vol.1, 2002

[BRI90] J. S. Bridle. "Probabilistic interpretation of feed forward classification network outputs with relationships to statistical pattern recognition". In F. Fogelman Souli'e and J. Hérault, editors, *Neurocomputing: Algorithms, Architectures, and Applications*, pages 227--236. Springer Verlag, New York, 1990

[GBK06] S. Gritzalis, P. Belsis, S.K.Katsikas, Interconnecting Autonomous Medical Domains: Security, Interoperability and Semantic-Driven Perspectives, *IEEE Engineering in Medicine and Biology*, 2006, IEEE Press

[GKK01] Gligor V. D., Khurana H., Koleva R. K., Bharadwaj V. G., and Baras J. S., "On the negotiation of access control policies", in *Proceedings of the 9th International Security Protocols Workshop*, Cambridge U.K., LNCS 2467 Springer, pp. 188–201, 2001

[GQ94] Gong L. and Qian X. "The complexity and composability of secure interoperation". In *Proceedings of the Symposium on Security and Privacy*, pages 190–200, Oakland, CA. IEEE Press, 1994.

[GRI06] Gritzalis S., "A Policy-ruled Knowledge Dissemination Architecture for Supporting multi-domain Secure Interoperation", *The eJournal for Electronic Commerce Tools and Applications*, Vol. 1, No. 4, 2006

[KABS06] Kaburlasos, V.G., "Towards a Unified Modeling and Knowledge-Representation based on Lattice Theory", *Computational Intelligence and Soft Computing Applications Series: Studies in Computational Intelligence*, Vol. 27

[KGL02] Khurana H., Gligor V. D. and Linn J. "Reasoning about Joint Administration of Coalition Resources", *Proc. of the IEEE International Conference on Distributed Computing Systems*, pp.429-439, Vienna, July 2002

[KS97] D. Koller, and M. Sahami, "Hierarchically classifying documents using very few words", in *International Conference on Machine Learning (ICML)*, pp. 170-178, 1997.

[LEW92] D. Lewis, "Feature selection and feature extraction for text categorization", Morgan Kaufmann, San Francisco, pp. 212-217, 1992.

[MAW05] R. Mukkamala, V. Atluri and J. Warner, "A Distributed Service Registry for Resource Sharing among Ad-hoc Dynamic Coalitions," *proc. of IFIP 11.1 & 11.5 Joint Working Conference on Security Management*, Fairfax USA, 2005.

[MLA98] D. Mladenic, "Feature subset selection in text-learning", in *Proc. of the 10th European Conference on Machine Learning*, 1998

- [MPB05a] Malatras A., Pavlou G., Belsis P., Gritzalis S., Skourlas C., Chalaris I., "Secure and Distributed Knowledge Management in Pervasive Environments", in *Proceedings of the 1st IEEE International Conference on Pervasive Services ICPS 2005*, V.Kalogeraki (Ed.), July 2005, Santorini, Greece, IEEE Computer Society Press
- [MPB05b] Malatras A., Pavlou G., Belsis P., Gritzalis S., Skourlas C., Chalaris I., "Deploying Pervasive Secure Knowledge Management Infrastructures", in *International Journal of Pervasive Computing and Communications*, Troubador Pub., vol. 1, issue 4, 265-276.
- [NWO02] Nejdl, W., Wolf, B., Qu, C., Decker, S., Sintek, M., Naeve, A., Nilsson, M., Palmer, M., Risch, T. "Edutella: A P2P networking infrastructure based on rdf". In: *Proceedings to the Eleventh International World Wide Web Conference*, Honolulu, Hawaii, USA (2002)
- [SAN00] Sandhu R., Ferraiolo D., and Kuhn R., "The NIST model for role-based access control: towards a unified standard", in *Proceedings of the 5th ACM Workshop on Role-Based Access Control (RBAC'00)*, pp. 47–63, 2000
- [SEL04] Seleznyov A., Mohamed A., Hailes S. "ADAM: An agent-based Middleware Architecture for Distributed Access Control" in *Proceedings of the 22nd International Multi-Conference on Applied Informatics: Artificial Intelligence and Applications*, 2004
- [SJB05] Shafiq B., Joshi J., Bertino E., Ghafoor A. "Secure Interoperation in a Multidomain Environment Employing RBAC Policies," *IEEE TKDE*, vol. 17, No. 11, pp. 1557-1577, Nov., 2005
- [SS99] R. Shapire, Y. Singer "Improved boosting algorithms using confidence-rated predictions. *Machine learning* 37(3): pp. 297-336, 1999
- [TEM04] Tempich C., Ehrig M., Fluit C., Haase P., Marti E.L., Plechawski M., Staab S. "XAROP: A Midterm Report on Introducing a Decentralized Semantics based Application", in *Proceedings of Practical Aspects of Knowledge Management (PAKM 2004)*, Vienna Austria, LNAI vol. 3336 Springer, pp. 259-270, 2004
- [WEI04] Weippl E., Schatten A., Karim S., Tjoa A. "SemanticLIFE Collaboration: Security Requirements and solutions – security aspects of semantic knowledge management", in *Proceedings of Practical Aspects of Knowledge Management (PAKM 2004)*, Vienna Austria, LNAI 3336 Springer, pp. 365-377, 2004
- [XACML] "Extensible access control markup language specification 2.0", OASIS Standard, (available at <http://www.oasis-open.org>), accessed May 2005
- [XPATH] www.w3.org/TR/xpath (Accessed May 2005)
- [YWJ05] C. Yao, W. Winsborough, Jajodia S., "A hierarchical Release Control Framework", proceedings of IFIP 11.1 & 11.5 Joint Working Conference on Security Management, Fairfax USA, December 2005.