2013 Ninth International Conference on Semantics, Knowledge and Grids

An Access Control Model for Organisational Management in Enterprise Architecture

Khaled Gaaloul[1] and H.A. (Erik) Proper[1,2]
[1]Centre de Recherche Public Henri Tudor, Luxembourg
[2]Radboud University Nijmegen, the Netherlands
{khaled.gaaloul,erik.proper}@tudor.lu

***Abstract*—Enterprise architecture (EA) aims to provide management with appropriate indicators and controls to steer and model service-oriented enterprises. EA offers a suitable operating platform to support an organisation's future goals and the roadmap for moving towards this vision. Despite significant research interest in the domain, common enterprises architecture frameworks lack of access control mechanisms supporting security requirements within organisations. Security has become a matter of paramount concern when managing organisations resources such as stakeholders' authorisation or sensitive data. In this paper, we propose an innovative approach for managing organisational resources in enterprise architecture. In doing so, we reason about task-based resources in the EA language ArchiMate. The idea is to build a conceptual model supporting access control when modelling a business process (set of tasks) in ArchiMate. We then map the common concepts with the role-based access control model (RBAC) to specify the required authorisation policies as part of the security specifications and guidelines in EA. Finally, a case study illustration will be used for the evaluation as part of the research approach.**

Keywords: Enterprise architecture, Access control, Task, Authorisation, ArchiMate, RBAC.

I. Introduction

Enterprise architecture is generally considered to provide a good steering instrument to analyse the current state of the enterprise, identify and describe alternative future states, guard the cohesion and alignment between the different aspects of an enterprise such as business processes and their ICT (Information and Communications Technology) support [1], [2].

The architecture modelling languages aim to support EA specification and description of enterprises components and their relationships; thereby ensuring an overall picture of the enterprise design and deployment. A prime example is the ArchiMate modelling standard for EA [3]. This means that architects can use ArchiMate to model, amongst others, an organisation's products and services, how these products and services are realised by business processes, and how in turn these processes are supported by information systems and their underlying IT infrastructure [3]. However, such techniques and languages do not address security issues in a satisfactory way [3], [4], [5]. For instance, access control artifacts are simply represented on the IT level without taking into accounts the process definition and the managing of organisations resources (e.g., stakeholders' authorisation, sensitive data).

Moreover, organisations establish a set of security policies, that regulate how the business process and resources should be managed [6]. Controlling access is considered by most information systems security experts to be a cornerstone to achieve information security. Access control, which may be physical, technical, or administrative, is a mechanism to provide information security. A given information system can implement access control systems in many places and at different levels. Operating systems use access control to protect files and directories while database management systems apply access control to regulate access to tables and views. Access control policies describe the ways in which information is managed and company assets are protected by mediation of access requests of principals or other information systems [7]. One of this standard is the role-based access control (RBAC) model [8].

The contribution of this paper is the improvement of an architecture language by leveraging access control aspects. To that end, we define a task-based access control model covering both the process definition and the resources allocation. The process definition is a composition of tasks where the aforementioned model will help us to distinguish the different concepts when modelling a business process in order to align it with the three-layer language ArchiMate. Further, the resources allocation will motivate the RBAC model to be mapped to ArchiMate and then specified in terms of security policies.

In light of these, the main research questions addressed in this paper can be summarised as follows: How to integrate access control concepts into EA models? In particular, how to integrate the RBAC concepts into ArchiMate metamodel and what is the impact on EA guidelines and regulations?

The remainder of this paper is structured as follows. Section 2 presents the research backgrounds. The task metamodel and the extension of ArchiMate with the RBAC concepts are explained in section 3. Section 4 is dedicated to the case study and its evaluation. Section 5 presents related work. Finally, section 6 concludes and outlines future work.

II. Backgrounds

In this section, we present the main ingredients building our approach. We start with the enterprise context and focus on a specific language offering a holistic view for organisations

978-1-4799-3012-8/14 $31.00 © 2014 IEEE
DOI 10.1109/SKG.2013.12

when modelling service-oriented architecture: the ArchiMate modelling language. We remind also about security requirements in organisations and introduce a standard dealing with role-based access control management: the RBAC model. Both models are chosen based on our research focus and are explained step-by-step in the following sections.

A. Enterprise Architecture

Enterprise Architecture (EA) is generally considered to provide a good steering instrument to analyse the current state of the enterprise (As-is), identify and describe alternative future states (To-be), guard the cohesion and alignment between the different aspects of an enterprise such as business processes and their ICT (Information and Communications Technology) support [3]. Architecture is a consistent whole of principles, methods and models that are used in the design and realisation of organisational structure, business processes, information systems, and infrastructure [9].

The unambiguous specification and description of components and especially their relationships in architecture requires a coherent architecture modelling language [9]. Current languages for modelling in the area of organisations, business processes, applications, and technology share a number of aspects on which they score low. For instance, the relation between domains is poorly defined, and the models created in different views are not further integrated. Besides, most languages miss the overall architectural vision and are confined to either the business or the application and technology subdomains [3].

B. The ArchiMate language

ArchiMate is an Open Group standard [10] for the modelling of enterprise architectures[1], emphasizing a holistic view of the enterprise. This means that architects can use ArchiMate to model, amongst others, an organisation's products and services, how these products and services are realised and delivered by business processes, and how in turn these processes are supported by information systems and their underlying IT infrastructure. Such a holistic perspective on an enterprise helps to guide change processes [9], provides insight into cost structures, and more [2].

The ArchiMate language defines three main layers [3]:

- The Business layer offers products and services to external customers, which are realised in the organisation by business processes (performed by business actors or roles).
- The Application layer supports the business layer with application services which are realised by (software) application components.
- The Technology layer offers infrastructure services (e.g., processing, storage, and communication services) needed to run applications, realised by computer and communication devices and system software.

The core concepts that are found in each layer of the language are depicted in figure 1. A distinction is made between structural or static aspect and the behavioural or dynamic aspect. Behavioural concepts are assigned to structural concepts, to show who or what displays the behaviour [3]. In addition to the active structural elements (business actors, application components and devices that display actual behaviour), the language recognizes passive structural elements, i.e., the objects on which behaviour is performed.

	Passive structure	Behaviour	Active structure
Business	business object	business services, functions and processes	actors and roles
Application	data objects	application services and functions	application components and interfaces
Technology	artifacts	infrastructure services and nodes	devices, networks and system software

Fig. 1. The Core Concepts of ArchiMate

In [9], the authors have compared a selection of standards and languages (e.g., RM-ODP, UML, BPMN and ARIS) to ArchiMate, using three criteria for comparison: frameworks, architectural viewpoints and domains that are covered by each language. According to their comparison, ArchiMate distinguishes itself from most other languages by its well-defined metamodel, concepts and, most importantly, its relations. The abstraction level of ArchiMate simplifies the construction of integrated models, where most languages appear to persuade architects to detailed modelling [9].

C. Access control management

A security policy defines the expected standard of security enforcement using access control within an organisation. Primarily, a security policy addresses who has access to what resources, as well as how this access has to be regulated and managed [11]. In most organisations, a security policy must be applied to hundreds, if not thousands, of employees. To simplify security administration, many organisations define roles with which multiple individuals can be associated. The security policy of the organisation then defines how permissions are to be associated with these roles.

Sandhu *et al.* presented the RBAC approach which is particularly effective when changes are made to the organisational security policy. The RBAC model needs only to be made to role assignments, which are significantly fewer than individual assignments [8]. Figure 2 the RBAC model with a set of users, roles, permissions and constraints. A user defines a human being. A role is a job function or a job title. Permission is an approval of executing (i.e. operation) an object (i.e. resource). A session is a mapping between a user and possibly many roles where it is associated with a single user (so-called a subject) and each user may establish zero or more sessions. Constraints restrict permissions depending on contextual information such as a conflict of interest [12].

D. Synthesis

Because of its inherent holistic nature, ArchiMate lacks specific guidelines for modelling an enterprise from a security perspective [3], [10], [4]. A security perspective focuses on access control management within an organisation, describing

[1] http://www.opengroup.org/archimate/

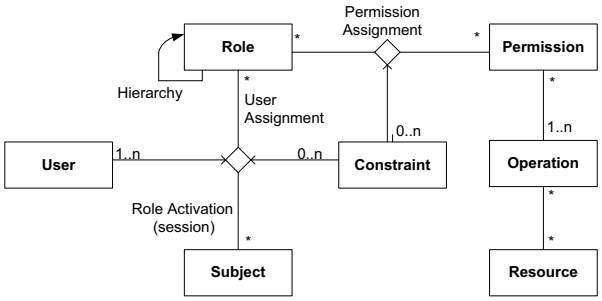

Fig. 2. The RBAC model

the role of each actor and the scope of his action when accessing organisation's resources. Such a perspective would nicely complement ArchiMate in the sense of providing security when managing organisational resources such as sensitive data.

III. Access Control in ArchiMate

In this section we propose to extend the core concepts of ArchiMate with the RBAC model. We define a task model based on workflow specifications. This inspiration is motivated by the reciprocity between ArchiMate and the workflow reference model [13], [14], [3].

A. A task based-organisational resources model

We explore an earlier work on the organisational resources analysis during the design time of a workflow model where resources and tasks are linked through a construct role [5]. We define a task as a set of applications or services that are accessed by subjects via specific functions. Figure 3 shows a metamodel for task-based resource model, which analyses the possible ways the resources access can be defined using RBAC specifications. During the process modelling, one task can generate several task instances. Note that we distinguish task type element from task since we assume that a task represents an instantiation of a task type during execution, equally for business objects. A task instance corresponds to an actual execution of a task. This specific execution of the task (a task instance) is allocated to only one subject through its unique worklist, where a subject defines a user selecting a role during runtime.

From a process perspective, a role is a subject to authorisations that define permissions (operations) for the execution of a task. A role represents a granted authorisation for a user (cf. the RBAC model). The design of the resource model can follow two different directions namely the material and human resources. Material resources define business objects and the way to use them where human resources define the actors.

From an application perspective, we define permissions as functions with operations to manipulate business objects (resources). We define a subject as an assigned user who is member of a role to claim a task instance. The task execution is added to the subject worklist. It defines the set of task instances claimed by this subject. The access to resources will be defined based on the authorisation instances to manipulate task's resources.

We aim to address issues related to the organisational management in enterprise architecture with regards to task's assignments and resource's access. We mainly focus on both process and application perspectives to analyse secure task requirements. We can identify from our analysis the relevant concepts and relationships to the ArchiMate metamodel:

- The set of *Tasks*, *Roles* and *Objects* are relevant the business layer in ArchiMate (see figure 1).
- The set of *Applications*, *Functions* and *Objects* define the realisation of the task assignment and are defined in the application layer of ArchiMate (see figure 1).
- The set of *Subjects*, *Worklists*, *Task Instances* and *Authorisation instance* defines the task execution requirements.

As mentioned before, this will lead us to investigate the organisational needs as well as the security requirements in ArchiMate. In doing so, we have identified the gap to fill between the security and the architecture domains: RBAC and ArchiMate respectively. In the following, we identify the relevant access control concepts and relationships to be mapped to the business and application layers of ArchiMate. This work has been facilitated by the application of existing approaches for ontology mapping [15].

B. Business layer and access control

Figure 4 describes the mapping between the ArchiMate business layer model and the RBAC model. Note that we use only an excerpt of the ArchiMate business layer metamodel to focus on these concepts and relations relevant for the access control model RBAC. The ArchiMate business layer metamodel concepts and relations are adapted from [9]. Concepts and relations mapping have been facilitated by the application of existing approaches for ontology mapping [15]. They are explained as follows:

- The concept *Business actor* defines an individual persons (e.g., customers or employees), but also groups of people (e.g., departments or business units) within the organisation. In RBAC, we define a *User* as a specialisation of a *Business actor*.
- The concept *Business role*: A role that an actor fulfills in an organisation. Importantly, this role is usually defined as the work carried out by an actor. In RBAC, we define an organisational *Role* as a specialisation of a *Business role*.
- The concept *Business object*: An entity manipulated by behavior such as business processes or functions. In RBAC, we define a *Resource* as a specialisation of a *Business object*.
- The RBAC relations: *User assignment* and *Permission assignment*, to manipulate resources, are respectively defined in ArchiMate as *assigned to* and *accesses* relations.

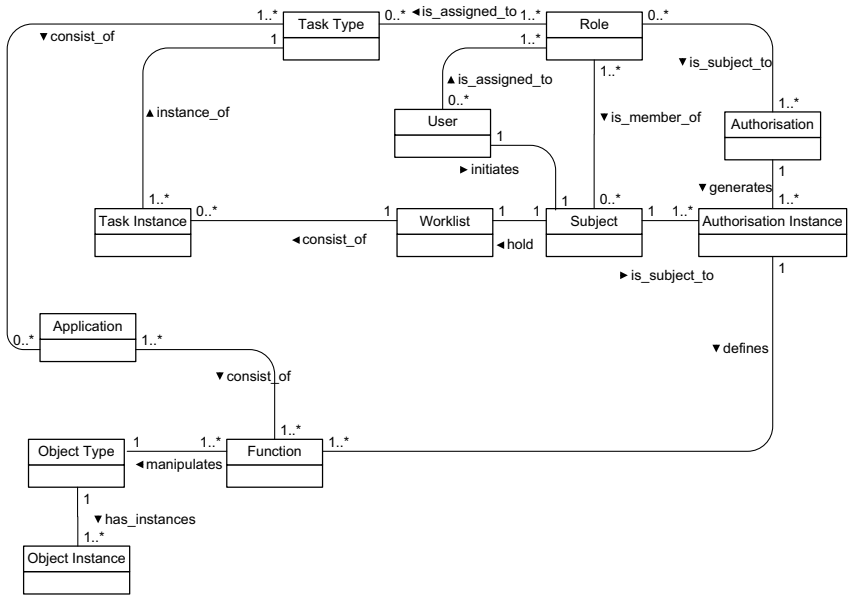

Fig. 3. A model for task-based organisational structures

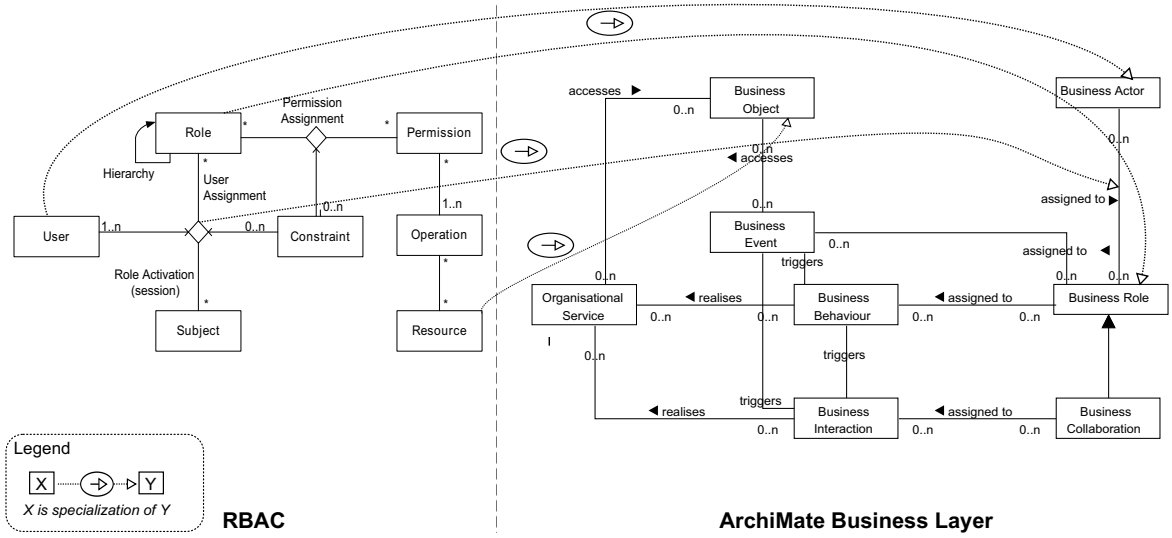

Fig. 4. Mapping of ArchiMate business layer with RBAC concepts

Note that an *Organisational service* is a unit of functionality that is meaningful from the point of view of the environment. Business processes, business functions and business interactions realise such a service. Moreover, A business process/function is a unit of internal behaviour, performed by one or more roles within the organisation. In this paper, we assume that a task is a business function composing a business process (see figure 3).

C. Application and access control

The Application layer supports the business layer with application services which are realised by (software) applications. The main structural concept for the application layer is the *Application component*. This concept is used to model any structural entity in the application layer [9]. In figure 3, the permission assignment related to authorisation is supported by the *Application* concept which defines operations to manipulate business objects. This represents the *Permission* concept in RBAC.

The access control management in RBAC is defined at the application layer via an application component such as software or server. To that end, it is necessary to interpret the business components such as the business role, the business actor or the permission at the application layer. The link between the business and application layers is the realisation

Task type	Role	Application	Function	Business Object type
T1. Receive Request	Prosecutor	CMS	read()	Request Document
T2. Check Request	Prosecutor	CMS	query(), update()	Request Document
T3. Translate Document	Prosecutor	CMS	translate()	Request Document
T4. Prepare Content	Assistant	CMS	add()	Request Document
T5. Send Request	Prosecutor	CMS	send()	Request File
T6. Review Request	Prosecutor	CMS	update()	Request File
T7. Determine Judicial Authorities	Judge	CMS	add(), modify()	Request File
T8. Forward Request	JAO	CMS	send()	Request File

TABLE I
RELATIONS BETWEEN TASK, ROLE, APPLICATION, FUNCTION AND BUSINESS OBJECT

relation in ArchiMate. This relation links a logical entity with a more concrete entity that realises it [3].

IV. ILLUSTRATIVE EXAMPLE

In this section, we briefly introduce a real case study from an e-Government project and identify relevant concepts for access control management. The goal is to evaluate an RBAC solution as an application component for ArchiMate and specify security policies as part of the security specifications and guidelines in enterprise architecture.

A. Mutual legal assistance scenario

We introduce an e-Governmental scenario related to the European administration within the areas of law enforcement and justice [16]. Mutual Legal Assistance (MLA) defines a scenario involving national authorities of two European countries regarding the execution of measures for protection of a witness in a criminal proceeding. The project infrastructure integrates systems such as the Case Management System (CMS) to access and process data.

In our example, we distinguish *Prosecutor* as the main responsible that collaborates with internal and external employees (Assistant, National Correspondent (NC), Judge and Judicial Authority Officer (JAO)) to process the MLA request. First, *Prosecutor A* receives the request and checks it in the MLA information service (tasks 1, 2 and 3). If the provided information are correct, the *Prosecutor* will continue to process the request by asking for the preparation of the request document by his assistant (task 4). After the preparation of the required legal documents, the *Prosecutor* will send the request to his colleagues in country B (task 5). The next steps that need to be taken are the review of the request, the determination of the judicial authority in order to forward the request to the concerned authority in country B for the final approval (tasks 6, 7 and 8).

The supporting table I summarises the required roles, applications, functions and business objects associated to tasks.

B. The access control framework

The application CMS is defined to support data access and processing. At an architectural level, this application is defined at the application layer of ArchiMate. We develop an RBAC solution supporting CMS based on the eXtensible Access Control Markup Language (XACML) specifications [17], [18]. This specification defines a profile for the use of the standard XACML to meet the requirements for RBAC.

A security policy defines the expected standard of security enforcement using access control mechanisms [11]. An access control has to be defined to check the authorisation of the initiated user (i.e. the *subject*). From table 2, we can specify an XACML policies. For instance, a policy where the decision returns *Permit* for an actor member of role *Prosecutor* on the task T1 is :

```
<Poliy>
<target>[Prosecutor,read,Request
Document]
<rule>[Permit]
<C>[none]
</Poliy>
```

We present an access control framework (ACF) to support authorisation policies within enterprise architecture (see figure 5). ACF is defined as a set of software components which accept requests to access resources, analyse these against policies representing actual access rights to resources, and return a response based on this analysis. To illustrate the original architecture of an ACF, a request is issued by the requestor, which is received by the *Receiver* component in ACF. This is then sent to the *Analyser* component that queries policies stored in a policy database. A response is generated by the *Responder* component, which defines a decision (permit, deny, or not applicable) that is sent back to the requester. The application components are: the *Receiver*, the *Analyser* and the *Responder*, where the policy database refers to data object in the ArchiMate application layer.

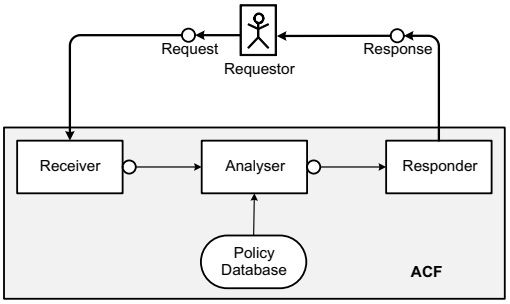

Fig. 5. Access control framework

C. Discussions

The application of access control management discussed in the previous section provides a partial solution to manage authorisations within EA. The work proposed in this paper covers the conceptual part of role-based access control in ArchiMate without validating it. Nevertheless, the change from the current state of the enterprise (As-is) to its future states (To-be) has to guard the cohesion and alignment between the different aspects of an enterprise such as business processes and their ICT. Dealing with organisational change, a security perspective focuses on dynamic access control management to be compliant with enterprise transformation. Ensuring compliancy will be conducted using security governance with regards to enterprise artefacts changes. This validation part will be submitted as a complementary research paper to this work.

V. Related Work

There exist several IT Governance frameworks that have some focus on enterprise security. One of the most known frameworks is the Control Objectives for Information and related Technology (COBIT) [19] which is already in the version 5 and has specific internal IT related goals with security (e.g., security of information, processing infrastructure and applications). One standard that focused on IT security is the ISO/IEC 2700 [20] which has a practice guide addressing access control issues. In the meanwhile, our work focuses on mainly the modelling aspect of enterprise architecture and the primer results provide access control concepts as well as policies in an enterprise architecture language.

In [4], authors presented an approach that enhances the ArchiMate standard with a responsibility modelling language for access rights management. The idea consists of aligning the business layer and the application layer of ArchiMate to ensure that applications manage access rights consistently with enterprise goals and risk tolerances. The alignment is realised by using the responsibility of the employees where the main focus of the alignment is the definition and the assignment of the access rights needed by the employees according to business specification. We differentiate from this work in mainly two aspects: (1) Our approach is driven by task's resources requirements and offers an objective representation of the business process in the different levels of an enterprise. (2) The identification of relevant concepts to the RBAC model helps to specify security policies as an important part of the organisation regulations.

The Zachman framework [21] is an enterprise architecture framework for enterprise architecture, which provides a formal and highly structured way of viewing and defining an enterprise. The framework defines six different perspectives (Scope, Business model, Information system model, Technology model, Detailed description and Actual system) describing the information which is considered essential in an enterprise architecture. These perspectives should be described in six different ways (Data, Function, Network, People, Time and Purpose). The Open Group Architecture Framework (TOGAF) is a framework for enterprise architecture which provides a comprehensive approach for designing, planning, implementing, and governing an enterprise information architecture [10]. TOGAF contains an architecture development method (ADM) that describes which steps should be taken to develop an enterprise architecture that has the four architectural domains (Business, Data, Application and Technology). Nevertheless, the Zachman framework and TOGAF do not provide any modelling methodology for constructing enterprise architecture.

VI. Conclusion and Future Work

In this paper, we have proposed an access control model supporting organisational management in enterprise architecture (EA). The novelty of this approach consists in bridging the gap between both access control and EA models. In doing so, the core concepts of the two domains were identified and defined. The main relations between concepts were analysed in order to construct the integrated conceptual model. The evaluation stage was done through a case study in the e-Government sector. The conceptual model was then deployed on an access control framework in order to specify security policies supporting the organisation regulations.

We believe that a step in future research can be represented by adopting this model to the whole EA framework by extending the three-layer in ArchiMate and including additional features (e.g. RBAC constraints). Another challenging topic will be the auditing and evaluation of security policies during the TOGAF Architecture Development Method (ADM) lifecycle.

References

[1] M. Op 't Land, H. Proper, M. Waage, J. Cloo, and C. Steghuis, *Enterprise Architecture – Creating Value by Informed Governance*. Springer, Berlin, Germany, 2008.

[2] M. Lankhorst, H. Proper, and H. Jonkers, "The Architecture of the ArchiMate Language," *Enterprise, Business-Process and Information Systems Modeling*, pp. 367–380, 2009.

[3] M. M. Lankhorst, *Enterprise Architecture at Work - Modelling, Communication and Analysis (4. ed.)*, ser. The Enterprise Engineering Series. Springer, 2013.

[4] C. Feltus, E. Dubois, E. Proper, I. Band, and M. Petit, "Enhancing the archimate standard with a responsibility modeling language for access rights management," in *Proceedings of the Fifth International Conference on Security of Information and Networks*, ser. SIN '12. New York, NY, USA: ACM, 2012, pp. 12–19.

[5] K. Gaaloul, "A Secure Framework for Dynamic Task Delegation in Workflow Management Systems. Ph.D. thesis, The University of Henri Poincaré, Nancy, France," 2010.
[6] V. Atluri and J. Warner, "Supporting conditional delegation in secure workflow management systems," in *SACMAT '05: Proceedings of the tenth ACM symposium on Access control models and technologies*. New York, NY, USA: ACM, 2005, pp. 49–58.
[7] F. I. S. M. A. FISMA, "The 2002 Federal Information Security Management Act (FISMA)," May 2002.
[8] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-based access control models," *IEEE Computer*, vol. 29, no. 2, pp. 38–47, 1996.
[9] H. Jonkers, M. Lankhorst, R. v. Buuren, S. Hoppenbrouwers, M. Bonsangue, and L. Van der Torre, "Concepts for Modeling Enterprise Architectures," *International Journal of Cooperative Information Systems*, vol. 13, no. 3, pp. 257–288, 2004.
[10] *The Open Group – TOGAF Version 9*. Van Haren Publishing, Zaltbommel, The Netherlands, 2009.
[11] C. P. Pfleeger and S. L. Pfleeger, *Security in Computing (4th Edition)*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2006.
[12] R. A. Botha and J. H. P. Eloff, "Separation of duties for access control enforcement in workflow environments," *IBM Systems Journal*, vol. 40, no. 3, pp. 666–682, 2001.
[13] WFMC, The Workflow Management Coalition, "Workflow Management Coalition Terminology and Glossary," 1999, document Number WFMC-TC-1011.
[14] ——, "Workflow Security Considerations," 2001, white Paper, Document Number WFMC-TC-1019.
[15] N. Noy and M. Musen, "The prompt suite: interactive tools for ontology merging and mapping," *International Journal of Human-Computer Studies*, vol. 59, no. 6, pp. 983–1024, 2003.
[16] T. A. R4eGov, "Towards e-administration in the large," March 2006, http://www.r4egov.eu/.
[17] Tim Moses, "eXtensible Access Control Markup Language (XACML) Version 2.0," 2005, committee specification, OASIS.
[18] K. Gaaloul and F. Charoy, "Task delegation based access control models for workflow systems," in *I3E 2009: Proceedings of Software Services for e-Business and e-Society, 9th IFIP WG 6.1 Conference on e-Business, e-Services and e-Society, Nancy, France, September 23-25, 2009.*, ser. IFIP, vol. 305. Springer.
[19] I. G. I. ITGI, *COBIT 4.1*. ISA, 2007.
[20] ISO/IEC, "ISO/IEC 27002: Information technology Security techniques Code of practice for information security management," 2005.
[21] J. Zachman, "A framework for information systems architecture," *IBM Systems Journal*, vol. 26, no. 3, 1987.