A Secure Public Sector Workflow Management System

Stefano Crosta*, Frederic Montagut*°, Jean-Christophe Pazzaglia*,
Yevgen Reznichenko°, Maarten Rits° and Andreas Schaad°

*Institut Eurécom
2229 Route des Crêtes - BP 193 -06904
Sophia Antipolis - France
Tel: (+33) 493 00 26 78 - Fax: (+33) 493 00 26 27
{Stefano.Crosta, Jean-Christophe.Pazzaglia}@eurecom.fr

°SAP Research
805, Avenue du Docteur Maurice Donat,
Font de l'Orme, 06250 Mougins
+33 (0)4 92 28 62 00
{frederic.montagut, yevgen.reznichenko,maarten.rits,
andreas.schaad}@sap.com

ABSTRACT

In this paper we analyze secure access control and rights management concerns in a typical public sector Workflow Management System which orchestrates the control flow of an inter-European judicial process. We have classified a set of topics, that have not been adequately addressed so far, in our opinion, in three different categories: i) deriving consistent access control policies for workflow tasks, ii) the temporal (short-term) provisioning of access rights with certificates, and iii) enforcing access control on workflow tasks, with a focus on inter-organizational workflows. We will analyze these different concerns in this paper, and propose specific solutions where appropriate. We have validated our work in a case study, closely related to the scenarios developed within the eJustice project, concerning an inter-organizational workflow regarding the issuing of rogatory letters and arrest warrants for the improvement of inter-European investigations and prosecutions[1].

Categories and Subject Descriptors

Access control, Middleware and distributed systems security, Security engineering and management

Keywords

Workflow security, Rights management, Attribute Certificate, Justice

1. INTRODUCTION

Performing complex business processes with the help of automated workflow systems will comprise the execution of different tasks. Each of these tasks might have to access specific and potentially sensitive data. Both the execution of tasks and the access to data should be seen as sensitive operations, to which access needs to be controlled.

Workflow Management Systems (WfMS) typically involve multi-layered application landscapes where access control enforcement is spread over the application stack (e.g. workflow layer, application server; database engine). In practice authorization exceptions tend to occur during execution of workflow tasks because the user's access rights are insufficient to satisfy the access control policies in the different layers. Common bad practice is to solve authorization exceptions by extending the rights granted to an organizational role without further in-depth analysis of why the authorization was denied. Performing such an analysis for every case is indeed difficult and time consuming, but these uncontrolled right assignments lead in the end to over-privileged accounts. Deriving consistent access control policies across the different application layers is thus one major concern.

WfMS implementations differ in the architecture and in the manner to interact with users and remote applications. With respect to security, our main challenge is the interoperability between different security models and different access right structures used in the applications triggered by the workflow engine. To avoid abuse of access rights, we encourage a short-term (temporal) provisioning of access rights. These rights may be embedded in a self-contained structure, a specific kind of certificates, generated on purpose and able to be interpreted by the different security modules using *ad hoc* wrappers.

A task is the natural granularity to describe atomic operations within workflows. Tasks will often trigger a remote process and therefore the workflow designer should provision the necessary rights to enable workflow execution. Defining fined-grained rights management policies for workflows is a major issue, yet assuring their enforcement during a workflow instance is as crucial; it is thus our aim to define a complete security architecture, encompassing a policy enforcement point enabling access control management at workflow task level.

Most of the observations in this paper are based on our insights into judicial information systems dealing with sensitive information which are subject to strict access control policies. Enforcing judicial processes through workflow systems makes multi-layered and interoperable access control mechanisms compulsory.

[1] This work has been performed in the context of the EU FP6 project eJustice: www.ejustice.eu.com

The paper is organized as follows. In section 2 we discuss related work. Section 3 presents an overview of the case study supporting our rights management analysis. In section 4 we discuss this analysis more in detail, focusing on the three topics that we briefly presented in the introduction. Section 5 explains the concrete implementation of the rights management architecture. Section 6 provides a summary and discussion of further work.

2. RELATED WORK

With the broad adoption of WfMS to orchestrate data exchange and activity coordination between organizations, security became a crucial and essential topic. Since the original and simple model presented in [24] many secure workflow models have been later developed. In [11] a whole framework to cope with integrity, authorization and availability issues within the workflow is introduced. The foundations of access control in inter-organizational workflows and their dynamicity are presented in [13].

In this paper we tackle three points of the workflow security research field: deriving consistent access control policies taking into account separation of duties constraints, just-in-time access rights provisioning with attribute certificates and access control policy enforcement on workflow instances and tasks. There is surprisingly not much prior research on the consistency of access control policies and especially the separation of duties constraints within the workflow. In [2] a solution is presented to provide access to workflow tasks only during their executions. Other approaches focus on the separation of duties during a workflow instance. Adaptive RBAC models [3, 20] applied to workflow objects and operations are presented in [7]. The conflicting entities administration paradigm CoAP, e.g. creating an order is conflicting with approving an order, is used in [5] to derive consistent policies The ideas presented in these articles exhibit similarities with our approach; however the granularity of the policy definition remains at the workflow abstraction level (e.g. per activity) without considering underlying methods and procedures.

There are two main categories of implementations of access rights management: centralized, using Access Control Lists (ACLs), and distributed, through Credentials. Different models exist to fine-grain access rights management; for simplicity in this paper we rely on the typical RBAC model, but can adapt to any other. The requirements of inter-domain, distributed systems such as the ones deriving from cross-European justice scenarios naturally lean towards the distributed model; it is still not common practice, though, to exploit credentials as all rounded authorization tokens in nowadays WorkFlows Access Control solutions.

Three main categories of certificates can be considered as possible credentials: Identity Certificates, linking a person's identity with his public key (e.g. X.509 [18] certificates), mainly addressing authentication; Authorization Certificates (e.g. SPKI [19]) binding public keys to access rights; and Attribute Certificates [12, 25] associating identities with authorizations. We propose an extended Attribute Certificate category, which we will describe more in detail in section 4.2. We present an application of these certificates in order to secure the access to cross-organizational resources, as well as in the access to workflow relevant data and for task execution.

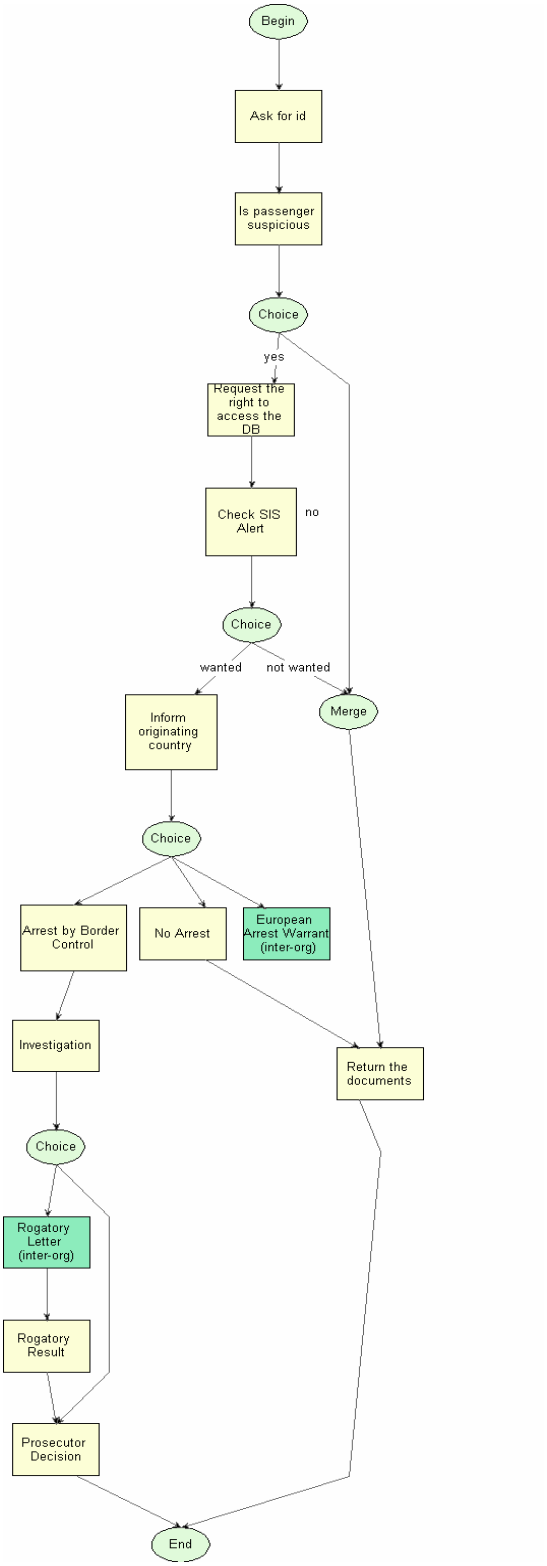

Figure 1: Border Control Scenario

3. CASE STUDY

In this section, we provide an example of one of the scenarios used to capture rights management requirements within the eJustice project. A prototype implementation of this scenario which uses our rights management framework will be described in more detail in section 5.

We consider the scenario as presented in figure 1. During a border control, a police officer checks the passport of passengers. For that reason, he will check the identity document (passport, national ID card, etc) presented by the person. We assume that a person will always be in possession of a document that satisfies the border control requirements.
If the officer has reasons to believe that the person is suspicious, he will submit an enquiry to the Schengen Information System (SIS)[2]. In absence of an existing alert no further investigation will be performed and the document will be returned. If there is an alert, the alert-originating-country has to be informed. Then there will be either an arrest by the border control officer, no arrest, or a European Arrest Warrant may be issued by a national judicial authority to require the arrest and return of the person[3]. The latter would lead to an inter-organizational workflow; the originating country has to launch a judicial procedure in their administrative domain. We will, however, not explore the legal framework of this procedure. If the border control officer arrests the individual, a further investigation has to be performed. During this investigation the legal authorities in the current country can obtain more information from the originating country by using the rogatory letter mechanism, also leading to an inter-organizational procedure. Based on the results of the investigation, the prosecutor will decide about the final outcome.

4. ANALYSIS

We will now present an analysis of the different access control requirements that appear in the scenario: i) deriving consistent access control policies for workflow tasks (section 4.1), ii) the temporal provisioning of access rights with certificates (section 4.2), and iii) enforcing access control on workflow tasks and associated data, with a focus on inter-organizational workflows (section 4.3).

4.1 Deriving Consistent Policies

In general, most applications are implemented as services on top of one or more databases. Access control enforcement is both situated at the application (method-) level and at the database level. Whereas access control policies of a service are defined independently with regard to the database policies, authorization mismatches are likely to occur in practice. It is common that database security policies are not finely tuned, and connections often established under over-privileged accounts, in order to avoid administrative overhead,. Therefore, the access control responsibility is shifted to the higher service layers introducing potential backdoors for malicious users. Workflow systems orchestrating these applications ntroduce a ‘virtual’ third layer. Access control in this layer is specified and enforced on workflow tasks.

Also, in real-cases, applied access control policies in the underlying layers are kept very permissive, because, if a fine-grained access control policy would be applied in the underlying application- and data-layers, then authorization exceptions are likely to occur during execution of workflow tasks, because the rights to access all underlying services are required. Our experience suggests that too often, access control policies in different layers are defined independently of each other. Moreover, authorization exceptions are in practice often solved by just extending the rights of the organizational role that has been assigned to execute the workflow task, without further in-depth analysis of why the authorization was denied. Performing such an analysis for every case is, in practice, not easily feasible. These uncontrolled right assignments lead in the end to over-privileged system accounts.

In [17] we proposed a methodology that allows specifying consistent access control policies. We developed a semi-automated tool called eXtreme access control Tool (XacT) based on aspect-oriented programming techniques [14]. This tool enables us to derive the organizational roles, if any, that have the required rights to execute all of the underlying methods in a workflow task. Alternatively, we can also derive the minimal set of requested access rights to perform a certain task, instead of retrieving the organizational roles that have at least this minimal set of rights assigned. This approach enables us to support fine-grained context-based access control on each layer, because we detect authorization exceptions on higher layers. If no organizational role exists that contains all the requested rights, then the workflow administrator could split-up the task in question into more than one task. Different roles can then be assigned to each of these new sub-tasks, with each role containing the requested rights for these sub-tasks.

The tool thus provides the administrator of the workflow with a ‘recommendation’ for assigning an organizational role to a task in the workflow. But, additionally, it can also calculate the rights that should be granted to be able to perform the task without causing authorization exceptions. These rights are then embedded within attribute certificates that will be discussed in next sections.

---

[2] The Schengen Information System is a shared database that uses a computerized system to place alerts concerning persons or property at the disposal of the authorities of each Member State.

[3] The person whose return is sought should be accused of an offence for which the maximum period of the penalty is at least a year in prison, or if he or she has been sentenced to a prison term of at least four months [10].

4.2 Managing and Delegating Rights

By essence, judicial procedures deal with sensitive information. The right to access information (e.g. evidences, etc) and to take initiatives (e.g. investigation, etc) is derived from a complex normative structure rooted in our democratic system and legislative infrastructure. In order to protect citizen rights in term of privacy and protection of data [8], but also to insure transparency, our rights management system should at the same time enable fine-grained access control and full traceability for authorized administrators. We should therefore be able to trace not only the identity of the user, but also to identify the chain of command (ministry, court, case) and potentially the context (terrorism threat) exhibited to enter a workflow or to access some information. Moreover, cross European scenarios raise this level of complexity since authorization rules are defined by multiple organizations involving different structures and security policies without supranational authority. We will show how our flexible Attribute Certificates can provide a pragmatic solution to express mutual trust and delegation of rights, while supporting consistent policies in a distributed environment, thus satisfying the given requirements.

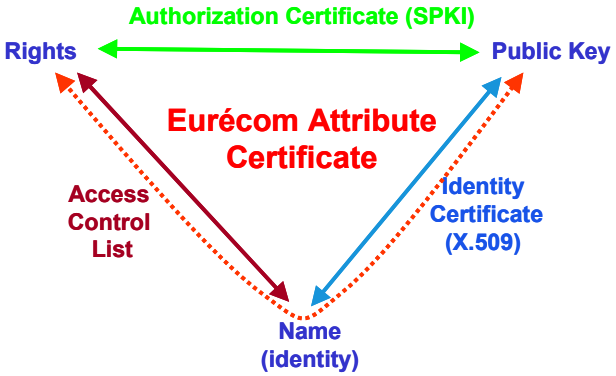

Figure 2: Eurécom Attribute Certificate

4.2.1 Eurécom Attribute Certificate

While certificates propose a valid alternative for distributed access control and identity management, existing solutions were not found which completely satisfy our requirements in terms of flexibility, traceability and advanced delegation. (

The Eurécom attribute certificate model (EuréCA, [12]) places identity and attributes on the same level, and allows embedding information to assess platform trust, restrict rights to resources, embed roles, etc. All attributes are defined in a uniform and extensible manner, with "identity" being just one like any other attribute.

EuréCA certificates provide a flexible data structure for the management of distributed credentials. A certificate associates attributes to a principal, the holder, with a data structure signed by the certificate issuer; Certificate holder and issuer can either be a public key, like in SPKI, or a reference to another certificate (for example X.509v3 or EuréCA certificates). Using

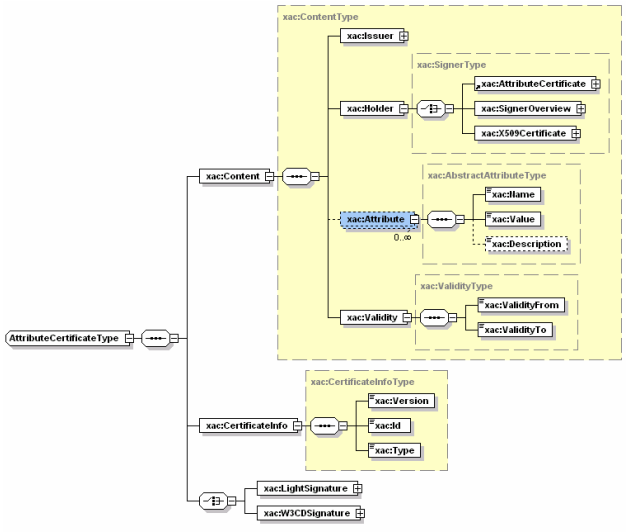

Figure 3: EuréCA XML Schema

a public key as Holder creates a new, independent attribute certificate from the Issuer for the Holder; using a reference to an existing certificate permits to associate attributes to existing entities, and can be used to build cross-domain trust chains.

Attribute structures traditionally associate a *name* with a *value*. As an open framework, the EuréCA model promotes the use of polymorphic attribute values: specific xml schemas can be developed to capture application[4] level semantic both in term of structure and behavior: the structural aspect can be used to further refine the value of an attribute, while behavioral aspect enables to add, for example, complex delegation rules.

A library providing support to these certificates is providing, including easy access to certificate values and management of certificates (generation, delegation and validation). Since the semantic of an attribute cannot be known at the certificate API level, the content of the attribute must be evaluated at the application level. Nevertheless, the library exposes a plug-in based architecture, which permits to easily develop new attribute types along with their validators, and provides built-in support for basic rules. Moreover, these validators are likely to use validation rules (such as delegation) using data embedded in the attribute itself, thus providing certificates which carry along all the information necessary for their validation, the library being just a ready-to-use implementation assisting developers.

EuréCa certificates are designed to embed the issuer certificate. exploiting recursive XML schema capabilities. While this increase size of certificates over any delegation iteration, it also permits to build a trust chain which allows for completely distributed validation; EuréCa certificates are fully self-contained. The Holder proves the ownership of the certificate through a classic challenge-response protocol demonstrating private key ownership.

The opportunity for any entity to act as a certification authority makes this framework flexible and particularly suitable to distributed organizations aiming to trace the chain of command.

[4] This type of attribute will extend the ComplexType AbstractAttributeType as shown in Figure 3.

The Issuer may create new attributes or delegate existing attributes, according to the delegation rules; he may also further reduce the right to delegate.

In the following two sub-sections we will highlight the need for the delegation of access rights to either execute external applications or to be able to access workflow relevant data.

4.2.2 Execution of External Applications

Different approaches may be envisaged to control the invocation of external applications, residing outside of the organizational boundaries, during the execution of a workflow.

A first, and rather primitive, approach is a loosely coupled solution where the engine limits itself to distribute tasks to users. Within this framework, obtaining necessary access rights will be entirely under the responsibility of the user. For this purpose, each external application will have to provide a dedicated authentication mean and relevant authorization to perform the different tasks. Finally, users will be responsible to inform the workflow management system to provide feedback on the task completion status. This kind of solution is often deployed to enable the collaboration of legacy systems. From a security perspective, it is often coupled with poor practices where login and password are shared between users with damaging effects on traceability.

A more advanced approach is to provide a distributed identity management system enabling a more tightly coupled collaboration between the workflow engine and the applications. Such solution must enable to provide, and enforce, a contract based agreement between the different organizations involved. In this scheme, the workflow management system, or associated services, will provide the necessary information to insure the effective implementation of the contract. Ultimately, the application may directly provide a feedback, enabling to resume the execution of the workflow. We argue that this typology of solutions is mandatory to promote a trustworthy collaboration environment between business partners dealing with sensitive or classified information.

With respect to our scenario (cf. section 3), the task 'Check SIS Alert' clearly involves the collaboration between the Border Control Department and the SIS Organization. Different security mechanisms (organizational, infrastructure and software) should impose drastic security measures and several security requirements can be foreseen:

1. Respective (and revisable) trust should be established between SIS and the Border Control department.
2. Interrogation of the system should be restricted to a small number of persons.
3. Interrogation should be performed by a qualified workflow engine.
4. Interrogation should be possible only when certain conditions are respected (time, presence of suspect, etc).
5. The Border Control Department, but also the SIS organization, should be able to trace precisely the requests, in order to track possible malicious behaviors.

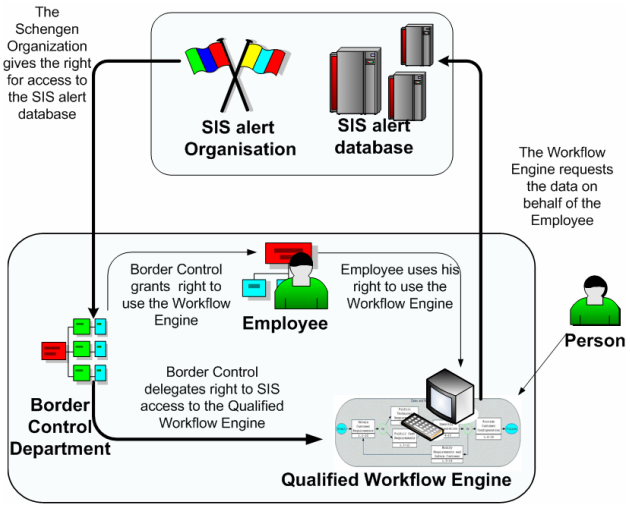

Figure 4: Access Right Delegation Path

Enforcing these requirements imposes to provide a software solution able to capture and to implement these measures. We will therefore describe our organizational choice conform to these requirements and sketch how EuréCA certificates[5] can be used to support it.

In the initial stage the SIS Organization, responsible to provide access to the SIS alert database, will grant a temporary (weekly, monthly) right to the Border Control Department to enable the interrogation of its system. A specific certificate embedding the prerogative to access the database and the right to delegate this prerogative will be issued by the SIS Organization to the Border Control Department.

However, to mitigate the risk of malicious insiders, the SIS Organization will require that the Border Control Department uses strong and certified means to authenticate its employees in charge of the control. This certification will be ideally done by an external and qualified certification authority. It is likely that an x509 certificate with the associated private key will be issued in a tamper proof Identity Card (PKI-enabled Smartcard) for this purpose. The Border Control Department will use this card as a standard authentication mean; however this certificate will only be used to grant temporary access to the workflow terminal operated at the checkpoints. Each day, during shift enrolment, a dedicated certificate will be issued for the duration of the shift using the identity certificate as Holder of the certificate[6]. This last certificate ('shift' certificate) will enable the employee to use the workflow engine to check persons.

To enable to run this workflow, the Border Control Department should therefore qualify a workflow engine. It is not our aim to discuss the qualification procedure but we may imagine that it will rely on code inspection and periodical audit of the hardware and software configuration. Once the qualification achieved, the Border Control Department will delegate the right to access the

[5] Unlike explicitly stated, certificate will refer to EuréCA certificate in this section.

[6] The implementation is using an intermediate EuréCA certificate to assess the organizational role of the person, however this does not significantly change our description.

SIS to the workflow engine in a specific certificate. It is interesting to highlight that, by choosing the validity duration, the SIS Organization will be able to force the Border Control Department to provide periodical audit without any administrative overhead.

Thanks to the provisioning of this last certificate, an employee will be able to check the SIS alert associated with a specific person. For this purpose, the workflow engine will issue a request providing a certificate using the 'shift' certificate as Holder and delegating the right to access the database. The figure 4 provides an outline of the solution.

Finally, it may also be interesting to add further attributes to this certificate to ensure the context of the control. This may for example involve a cryptographic proof of the presence of the controlled person to avoid fake investigation, of course supported by cryptographic hardware. This proof may also be used to ensure that the person is aware that a request has been performed against the SIS to comply with local privacy legislation. Such capability can be easily added assuming that an electronic ID with digital signature capabilities is available.

4.3 Fine-grained access-control to Workflow Relevant Data

During execution of workflows, process decisions and other control operations are based on a subset of data, called "workflow relevant data"[25]. This data set may be generated or updated by workflow application programs and may also be accessed by the different users participating in a specific workflow instance. Securing a workflow could therefore not be envisaged without controlling access to workflow relevant data. We introduce the new term of "task relevant data" as a subset of workflow relevant data, which is necessary for the execution of a certain task. Task relevant data is usually expressed as parameters for a task. In conventional Workflow Management Systems, the workflow engine sends the task relevant data together with the new task notification to users, thus they have direct access to it during the execution of a task.

4.3.1 Task relevant data access control

With respect to section 3, task relevant data in this scenario are, for example, the data concerning the suspect that are used in the border control investigation (identity, SIS alert, etc.) Since the tasks in the presented workflow are assigned to roles, the conventional WfMS would inform all employees assigned to the specified roles about the new tasks, and disclose possibly confidential task relevant data to all users assigned to the role. For instance, in the workflow during the Investigation task, every employee with the "officer" role would have access to the identity of the suspect and according SIS alert, although only one officer will really execute the task. To avoid this, we introduce the concept of the "Workflow Engine Guardian" which is presented in more details in the next section. The guardian is an additional software-component which enforces access control to running workflow instances and their workflow relevant data (as shown in Figure 6). The guardian is an intermediate between the workflow engine and the users in the system. The workflow engine passes new task notifications to the guardian, together with data specified as parameters for the task (task relevant data). The guardian extracts the task

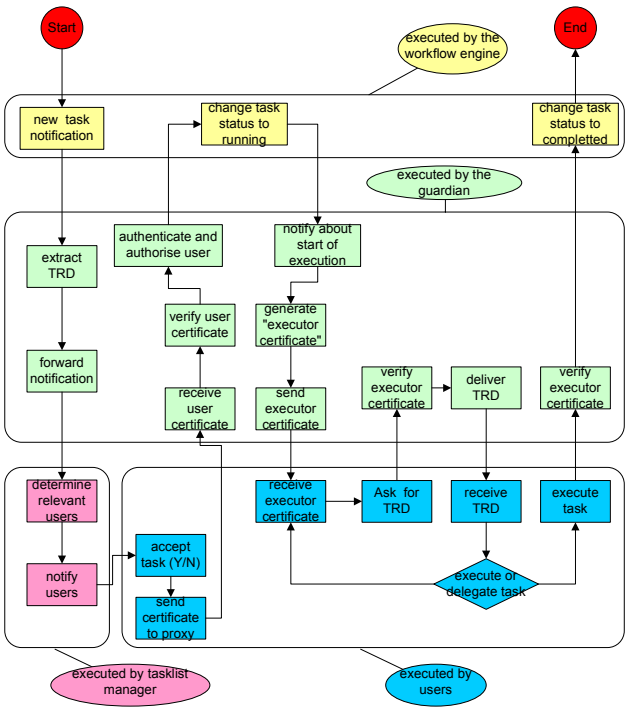

Figure 6: Task Execution Activity Diagram

relevant data and holds it back. Afterwards, it notifies all relevant users about the upcoming task and determines the real executor for the task. In order to retrieve the task relevant data, a user has to provide an attribute certificate, which certifies that he is the executor of the task. The exact procedures for the task assignation and retrieving of the attribute certificate are presented in section 4.4. After the user provides his attribute certificate, the guardian first verifies its validity. In case that the certificate is valid, the guardian processes the request further; otherwise it will log and discard it. In the next step of the procedure, the guardian has to verify if the access of the user to each task relevant data unit is compliant with the organization's security policy. In this case, the guardian sends all task relevant data to the user. Otherwise the set of task relevant data will be reduced by elements which the user is not authorized to access.

If we use XacT, as presented in section 4.1, the user executing the task should normally posses all the required rights. However, the organization's security policy is dynamic. For instance, we can verify that any officer during the Investigation task in the workflow presented in section 3 has the right to read the document containing the SIS alert. However, in a previous step in the workflow, another officer could have labeled one particular SIS alert as highly confidential. Therefore, it is possible that a specific user holding the role officer is not allowed to access this SIS alert any longer in the investigation task. The user originally assigned to the task could then delegate his attribute certificate, which certifies that he is executor of the task, to a new executor. To be able to execute the task, the new executor should hold either the same role as originally assigned to the task or a role with at least the same rights (e.g. the "captain" role which is higher in the role hierarchy). If this is the case and the new executor is authorized to access the data according to the organization's global security policy, then all task relevant data will be provided to the new executor.

Otherwise the task could be delegated again. A workflow instance and the tasks to be performed are processes that run under the control of the engine. We have to make sure that we can actually enforce the access control to these instances and tasks when either attribute certificates or just internal role-specifications are used, or both. The application will typically process this information using its security context. When we use attribute certificates, the additional verification methods that have been presented in section 4.2.2 have to be used.

4.3.2 Securing task dispatching

It is important to know that the workflow engine does not directly interact with users. Instead, the engine passes the data, which are required for performing the task, to a special application called tasklist manager (t-manager). The t-manager takes then action in order to notify the relevant users (as shown in Figure 6). Since the core competence of the workflow engine lies in the creation and management of workflow instances, administration of the workflow relevant data and notification about upcoming tasks [24], we decided not to implement security functionality directly into the workflow engine. Instead, an external security module maintains the task of securing the interaction between the workflow engine and users. We already introduced the guardian in section 4.2.2 in the context of access to task relevant data. All communication between the engine and the users is routed through this module.

Once the guardian receives a request for creation of a new workflow instance, it authenticates first the requestor and then checks if the user is authorized to create a new instance from the specified workflow model. If both procedures succeed, the request will be forwarded to the workflow engine, otherwise discarded.

At certain points during the execution of the workflow instances, some tasks need to be executed by the assigned users or applications. In this case the workflow engine generates a notification for each new task. The workflow engine passes these notifications to the guardian together with data specified as parameters for the tasks (task relevant data). In case that a particular task is assigned to a user or a role, the guardian will extract the task relevant data and hold it back. The idea of holding back the data has been explained in section 4.3.1. This makes especially sense if the task is assigned to a role and not to a particular user. After the new task notification has been sent out, the guardian waits for the commitment of the user to execute the task. It is the responsibility of the t-manager to find exactly one executor per task. When a task is assigned to a particular user, the t-manager simply needs to inform the user about the new task. However if no executor has been yet assigned, the t-manager needs to find an executor out of all users holding an appropriate role. As soon as the user is selected, he has to authenticate himself to the guardian. The guardian will then:

1. Assign the user as “effective executor” to the task.
2. Generate an Attribute Certificate for the user, allowing him to access the task relevant data and to complete the task execution. The certificate is then sent back to the user.
3. Change the status of the task in the workflow engine to “Running”.

Once the user begins the execution of the task he will certainly need to access the task relevant data held by the guardian. To retrieve the data, the user has to provide the attribute certificate, which was given to him in the previous step. The guardian verifies the certificate and checks if the user is allowed to access the data according to the organization’s security policy. We presented in section 4.3.1 a solution using delegation in case the user is not allowed to access these task relevant data. Once the user completes a task, he should notify the guardian. To get the notification accepted, the user should authenticate himself as holder of the certificate used for retrieving of the task relevant data. The status of the task in the workflow engine is then changed to “Completed”.

The case study in section 3 also contains inter-organizational processes as, for example, the European Arrest Warrant. At this point, the control flow leaves the Border Control Department and continues in the other organization (originating country). The proxies of the workflow engines running in the different organisations should be able to handle inter-organisational security aspect. As presented in [23], both organisations are participating in the coalition workflow with their workflow views. The workflow views are public interfaces to the organization’s private workflows. The business logic of the organization’s private processes is managed by these private workflows. The transfer of the control flow from one organization to the other causes status modifications of the workflow views and consequently of the private workflows of both partners. The last task in the private workflow of the Border Control Department (European Arrest Warrant) will change the status of the first task in the workflow of the originating country (not presented) from “not started” to “running” enabling to perform it. We can identify the following authorization requirements for this event:

- The organization transferring the control flow should be assigned to a special role in the coalition process. This role will be delegated by the workflow engine, at the task level, to the employee assuming that he may execute the task according to the local security policies. The role activation could be realized by issuing of special attribute certificates, which includes the role of the holder as an attribute.
- The transfer of the control flow should not violate the integrity of the coalition workflow status. This means that the control-transferring-partner should complete first prior tasks before activating the task in the other organization.

5. IMPLEMENTATION

In this section we conclude our analysis, as presented in previous sections, with two concrete implementation examples of some critical access control validation points in the presented use-case scenario. We consider consequently the execution of the external application containing the SIS alert database and the Investigation task in the Border Control workflow.

```
boolean checkSISAlertAccess(
   Context queryContext, User requestor){
1  CertificateValidator validator =
                  new  CertificateValidator();
2  validator.addTrustedCert(databaseCert);
3  validator.addTrustedCert(borderControlDepartCert);
4  Certificate officerECA = requestor.getCert(1);
5  Certificate engineECA = requestor.getCert(2) ;
6  String context = cert1.getAttribute("context");
7  boolean accesGranted=false;
8  if (validator.validate(officerECA) &&
9       (validator.validate(engineECA) &&
10    validator.validateAttributes(officerECA) &&
11    isContextValid(context) &&
12    hasAccessRight(officerECA){
13       accesGranted=true;
14}
15
16    logDBAccess(accesGranted ?"INFO":"WARNING",
         queryContext.get("personID"), requestor);
17 return accesGranted;
18}
```

Code Extract 1: SIS Alert Database Validation

The SIS alert database processes the request from the workflow engine concerning a particular person. As previously explained, in order to authorize the request, the database application has to perform several validations on the attribute certificates, as presented in Code Extract 1. Once the application providing the interface to the database receives a SIS alert request, it extracts the requestor certificates (line 4 & 5) and verifies if:

- The issuer chain of the certificates is valid (line 8 & 9). Since only the database's own certificate and the Border Control Department's certificate are trusted (lines 2 and 3), the validation will only succeed if the certificate was issued by any of these authorities.
- The attributes within the officer certificate are valid (line 10). They can only be validated if they were issued by a trusted authority. In addition, the delegation level of the attributes should decrease with each step on the delegation path.
- The context of the request is valid (line 11). As identified, in order to make a SIS alert request, the officer has to prove that he is in the context of a border control. A unique token has to be generated, signed by the officer and passed to the database. On the basis of this token the database application can verify if the person is in fact in front of the officer.

```
Vector getTaskRelevantData(Task task, User requestor){
1 // pdp represents the global security Policy
2PolicyDecisionPoint pdp =
               PolicyDecisionPoint.instance();
3
4 Vector taskRelevantData = task.getRelevantData();
5
6 Vector result = new Vector();
7 CertificateValidator validator = new
CertificateValidator();
8 validator.addTrustedCert(organisationCert);
9 if (validator.validate((requestor.getCert())) &&
10     validator.validateAttributes(
       (requestor.getCert())) &&
11             task.isValid() &&
12             task.isInExecution() &&
13     isExecutor(requestor, task)){
14     for i in  taskRelevantData{
15             if (pdp.isAccessAllowed(user, i))
16                     result.add(i);
17     }
18     }
19     return result;
}
```

Code Extract 2: Access Control to Task Relevant Data

The officer has the required access right (line 12). One of the certificate attributes should be the right for the access of the database.

Only if all of the above identified conditions are validated, the database will process the request (line 14) and send the result back to the workflow engine (line 18). For the later revision of accesses to the database the identity of the suspect and the requesting officer are logged (line 13).

As a second example, we consider the execution of the Investigation task, in which an officer tries to access the suspect's identity and the corresponding SIS alert. In this case the guardian has to verify if the officer has the right to access the task relevant data (suspect ID, SIS alert). The verification procedure is shown in Code Extract 2.

In order to get the access to the data, following conditions have to be fulfilled:

- The issuer chain and the attributes in the certificate should be valid (lines 9 and 10). Since the interaction occurs within the same organization, only the organization's own identity certificate needs to be trusted. The issuer chain is valid only if the top of the chain is the organization's (Border Control Department) identity certificate. The same is true for the attributes; they can only be validated if they are originally issued by the organization itself and the delegation rules are respected.
- The task for which the task relevant data are requested should exist and should be in execution (lines 11 and 12).
- Finally, the requestor should be assigned as "effective executor" to the task or this attribute should be delegated to him (line 13).

If all the conditions are valid, the guardian will reduce the set of task relevant data by elements that could not be accessed by the requestor (lines 14-17) according to the organization's global security policy. The requests concerning the organization's security policy are processed by a global Policy Decision Point (PDP) (line 15). This PDP manages the security policy of the organization.

6. SUMMARY & CONCLUSION

We provided in this paper an analysis of the rights management concerns in a typical public sector Workflow Management System. We have classified concerns that have not been adequately addressed so far, in three different categories: i) deriving consistent access control policies for workflow tasks, ii) the temporal provisioning of access rights (with certificates), and iii) enforcing access control on workflow tasks, with a focus on inter-organizational workflows.

We showed how using a distributed right management scheme based on Attribute Certificate merging identity for traceability and authorization can address these concerns. Our solution enables to support a high level authentication mechanism using existing certification authority infrastructure, to insure the provisioning of contextual and temporary access right to employee, to insure that the access is done using specific equipment and to keep track of the set of information which enabled the access. These characteristics can be achieved in a distributed environment and, since each step involves a digital signature, they can not be modified without being invalidated. Moreover, the effective implementation of auditing procedures can be enforced, and provision to add extra information to comply with local legislation can be envisaged.

This analysis and the current example of using the EuréCA and SAP Workflow tools in our scenario of border control revealed a number of future issues that require to be addressed. Our current delegation approach only focuses on granting (or delegating) some abstract access right to a principal using the EuréCA framework. We did so far not consider the semantics of the access rights and whether this might have an impact on the delegation scheme (e.g. to forbid delegation in case of a possible violation of separation of duty rules), nor detailed the impact on a standard RBAC model. Another question would the delegation of rights without owning them directly. We should consider an analysis of the type of attributes that we may want to embed in the EuréCA certificates using its extensible architecture.

7. REFERENCES

[1] Akenti : Distributed Access Control http://www-itg.lbl.gov/Akenti/

[2] Atluri, V. and Huang, W. *An Authorization Model for Workflows*. Lecture Notes in Computer Science 1146, 1996.

[3] Belokosztolszki, A. and Moody, K. Meta-Policies for Distributed Role-Based Access Control Systems. *In 3rd* IEEE Workshop on Policies for Distributed Systems and Networks, 2002.

[4] Bertino, E., E. Ferrari, and V. Atluri, *The Specification and Enforcement of Authorization Constraints in Workflow Management Systems.* ACM Transactions on Information and System Security, 1999. **2**(1): p. 65-104.

[5] Botha, R. A. and Eloff, J. H. P., *Separation of duties for access control enforcement in workflow environments*. IBM Sytems journals End-to-End Security, Vol. 40, No. 3, 2001, p 666.

[6] Damianou, N., et al. *The Ponder Policy Specification Language*. in *Policies for Distributed Systems and Networks*. 2001. Bristol: Springer Lecture Notes in Computer Science.

[7] Domingos, D., Rito-Silva, A. and Veiga, V. *Authorization and Access Control in Adaptive Workflows*. Proceedings of the 8th European Symposium on Research in Computer Security (ESORICS 2003), Springer-Verlag, LNCS, 2003.

[8] European Parliament and Council of Europe Directive 1995/46/EC, on the protection of individuals with regard to the processing of personal data and on the free movement of such data, October 24, 1995.

[9] European Parliament and Council of Europe Directive 1999/93/EC, on a Community framework for electronic signatures, December 13, 1999.

[10] EUROPA - European arrest warrant replaces extradition between EU Member States http://europa.eu.int/comm/justice_home/fsj/criminal/extradition/fsj_criminal_extradition_en.htm

[11] Hung, P. C. K., Karlapalem , K., *A secure workflow model*, in Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003 - Volume 21, pp. 33-41.

[12] L. Bussard, J. Claessens, S. Crosta, Y. Roudier, A. Zugenmaier - *Can we take this offline? Credentials for Web Services supported nomadic applications*
In Proceedings of the *4th Conference on Security and Network Architectures (SAR'05),* June 06-10, 2005.

[13] ICare project: http://www.cert-i-care.org/

[14] Kang, M. H., Park, J. S. and Froscher, J. N., *Access Control Mechanisms for Inter-organizational Workflow*, in Proceedings of the Sixth ACM Symposium on Access Control Models and Technologies, pp. 66-74, 2001.

[15] Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Videira Lopes, C., Loingtier, J.-M., and Irwin, J. Aspect-Oriented Programming. In Proc. of ECOOP, Springer-Verlag (1997).

[16] Morrie Gasser, Ellen McDermott. An Architecture for practical Delegation in a Distributed System. 1990 IEEE Computer Society Symposium on Research in Security and Privacy. Oakland, CA. May 7-9, 1990.

[17] Core and Hierarchical Role Based Access Control (RBAC) profile of XACML, Version 2.0. Committee Draft 01, 30 September 2004. OASIS Open.

[18] Rits, M., B. De Boe, and A. Schaad. XacT: A Bridge between Resource Management and Access Control in Multi-layered Applications. in ACM Software Engineering Notes of Software Engineering for Secure Systems (ICSE05),.May 2005. St. Louis, Missouri, USA.

[19] RFC 2527 - Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework: ftp://ftp.isi.edu/in-notes/rfc2527.txt

[20] RFC 2693 – SPKI Certificate Theory: ftp://ftp.isi.edu/in-notes/rfc2693.txt

[21] Sandhu, R. *Transaction Control Expressions for Separation of Duties*. in 4th Aerospace Computer Security Conference. 1988. Arizona.

[22] The Schengen Information System http://www.hri.org/docs/Schengen90/body4.html#chapter%201

[23] Schengen Information System II http://europa.eu.int/scadplus/leg/en/lvb/l33183.htm

[24] Schulz, K. and M. Orlowska, *Facilitating cross-organisational workflows with a workflow view approach.* Data & Knowledge Engineering, 2004. **51**(1): p. 109-147.

[25] WfMC (2001), Workflow Management Coalition (WfMC), *Workflow Security Considerations - White Paper*, Document Number WFMC-TC- 1019, Document Status - Issue 1.0.

[26] WiTness project: http://www.wireless-trust.org/