Policy Specification and Enforcement for Smart ID Cards Deployment

Ramaswamy Chandramouli
National Institute of Standards and Technology, Gaithersburg, MD, USA
mouli@nist.gov

Abstract

Deployment of Smart Cards for Identity Verification requires collection of credentials and provisioning of credentials from and to heterogeneous and sometimes legacy systems. To facilitate this process, a centralized identity store called Identity Management System (IDMS) is often used. To protect the integrity, confidentiality and privacy of the credential data that is collected, stored and disseminated through IDMS, a sophisticated set of policies governing data flows, processing and distribution are required. In this paper, we present a policy specification and enforcement framework using XML, XML Schemas and XSLT that was developed for secure management of the infrastructure system used for a large scale smart ID card deployment.

1. Introduction

Smart cards are now being increasingly used as personal identity verification tokens. This has become possible not only due to the fact that the costs of the smart cards have come down dramatically in the recent years but also due to the fact that smart card operating systems can support complex algorithms such as encryption/decryption algorithms and biometric matching algorithms. This capability now enables smart cards to store various types of credentials such as:

- A long personal identifier string (say 256 bytes) as compared to a 4-digit PIN that humans can remember and use, say, for ATM access.
- Biometric data such as fingerprint minutiae or digital facial image
- A signed digital certificate such Public Key Infrastructure (PKI) certificate and an associated Private Key.

To deploy smart cards for identity verification (let us call such types of cards as Smart ID cards) for multiple applications within an enterprise such as logical access to IT systems and physical access to facilities requires a sophisticated infrastructure for collection of credentials from multiple, heterogeneous and sometimes legacy systems and provisioning of credentials to such disparate systems such as individual IT access control systems or centralized IT access control systems such as Single-sign-on (generically called Logical Access Control Systems or LACS) and physical access control systems (PACS). To facilitate the management of multiple credential collection and provisioning data flows, a system called Identity Management Systems (IDMS) is often used. We name the IDMS used for smart ID card deployment as IDMS-SCD for where SCD stands for smart card deployment. The IDMS-SCD is a centralized data repository of all enterprise credentials and a software system that manages the storage, access and data flows (both in-flows and out-flows) of all enterprise credential data [1]. The IDMS-SCD thus forms the central point of trust in the entire infrastructure system used for smart ID card deployment and hence the most critical component.

The credential data coming in, stored and flowing out of IDMS-SCD, being used for identity verification, contains a category of data called Personally Identifiable Information (PII). The handling of PII is subject to several governmental privacy laws and regulations [2,3]. In addition, in order to protect the safety of individuals, the confidentiality and integrity of their identity information (credentials) must be protected. These confidentiality and integrity requirements not only cover the stored data in IDMS, but also apply to initiation of the various data flows in and out of IDMS. To cover these confidentiality, integrity and privacy requirements, therefore, requires a sophisticated set of policies governing the processing and generation of all credential data flows [4,5,6] and a flexible access specification framework that can represent and enforce these policies such as the one based on Role-based Access Control Model (RBAC) [7].

In this paper, we develop and present a policy specification framework for management of all credential data handled in an IDMS for supporting smart ID card deployment. The framework uses XML Schema for representing a role-based control model and domain constraints based on the model, XML for encoding of access specification and constraint data and XSLT transforms for encoding rules for validating access specifications and for privilege resolution.

The organization of the rest of the paper is as follows. In chapter 2 we describe various categories of information (credential flows) flowing in and out of IDMS-SCD and provide a detailed description of the policies governing the handling of each category of information. Chapter 3 describes the five components of our policy specification framework and provides examples of XML Schema, XML and XSLT encodings that are used in the implementation of the framework. In chapter 4, we summarize the characteristic features of our framework and discuss its benefits.

2. Credential Data Flows and governing policies for Smart ID Card Deployment

As already stated, in an IDMS used for supporting smart ID card deployment, credentials are collected from multiple sources and then provisioned not only to smart card production systems but also to enterprise applications (LACS & PACS) that will be authenticating users based on smart cards. The following are the categories of information flowing into and out of IDMS-SCD.

- Card Applicant Sponsorship Information
- Card Applicant Enrollment Information
- Card Issuance Approval & Card Production Information
- Physical Facilities Access Identity Information
- IT Systems Access Identity Information

Let us now look at the content of each category of information and policies associated with each category.

2.1. Card Applicant Sponsorship Information and associated policies

The information package contains mostly demographic (personal) information (such as Full Name, DOB, SSN, Country of Citizenship, Home Address, organizational unit(dept or division), Special Status (e.g., Emergency Official), Type of affiliation (Employee or Contractor) etc) about the card applicant (potential card holder) supplemented with information about the person who is sponsoring the card applicant. The latter information may consists of Sponsor's Personal Identifier, Sponsorship date etc. In addition, the Card Applicant Sponsorship information could consists of some information relating to the card that will be issued (Dual Interface Card with/without an additional barcode etc) and the address to which the card will be shipped for the card applicant to pick it up. As we could see, this category does contain some PII data such as Card Applicant's Full Name and SSN and Sponsor's Personal Identifier.

The policies governing the handling of this information category are:

- Sponsorship Information can be imported into IDMS only by users holding CardApplicant_Sponsor role (PO-1)
- A person holding CardApplicant_Sponsor role can only import sponsorship information pertaining to organizational units (department or division) that have been designated in his/her role indexing parameter (PO-2).
- A person holding the CardApplicant_Sponsor role cannot be assigned the Credential_Enroller or CardIssue_Approver role (PO-3)
- Since Sponsorship Information contains PII data, it must be transmitted in a secure manner from enterprise HR systems (from which it originates) to IDMS-SCD (PO-4).

Each of the above policies stated above has been formulated to incorporate some concepts or principles from the best practices for data governance. The underlying principles in each of the above policies are:

- PO-1 – Authorization Specification
- PO-2 – Principle of Least Privilege
- PO-3 – Avoiding Conflict of Interest (or Separation of Duty)
- PO-4 – Privacy, Confidentiality and Integrity Protection

The enforcement mechanisms for each of the above policies can take on different forms. The privacy, confidentiality and integrity protection for the transmitted information is ensured by setting up a secure VPN session and by encrypting and digitally signing the data content. To enforce all access related policies (authorizations, least privilege and avoidance of conflict of interest), an RBAC model is defined for IDMS using XML Schema. The authorization (access permissions) are specified in XML based on this IDMS RBAC XML Schema (let us call this RBAC-SCD schema where the abbreviation SCD stands for Smart Card deployment).

2.2 Card Applicant Enrollment Information and associated policies

This category of information is made up of data that forms part of authenticating biometric credentials such as fingerprint minutiae, digital facial image etc and identity proofing documents such as scanned images of passport, driver's license etc. Most of the information under this category are PII and must be subject to privacy protection.

The following policies apply to this category of information:

- Enrollment Information can be imported into IDMS-SCD only by users holding Credential_Enroller role (PO-5)
- A person holding Credential_Enroller role can only collect enrollment information from card applicants from the set of regions that have been designated in his/her role indexing parameter (PO-6).
- A person holding the Credential_Enroller role cannot be assigned the CardApplicant_Sponsor or CardIssue_Approver role (PO-7)
- Since Enrollment Information contains PII data, it must be transmitted in a secure manner from Enrollment Workstations (from which it originates) to IDMS-SCD (PO-8).
- Raw fingerprint data after it is converted into fingerprint minutiae data to be included in the Enrollment package should be deleted and no longer be retained at the Enrollment Workstation (PO-9).

2.3. Card Issuance Approval Information and associated policies

This consists of authorization information (for smart card production) directly entered into IDMS-SCD by a responsible official of the organization after the card applicant's identity has been vetted and background verified. Apart from authorization information, the official usually enters the email ID of the applicant to enable the system to generate a unique ID called UPN (to be used as the Account ID in the Corporate Directory) and also enters a Distinguished Name (DN) to be used as the identifier for generating an end-user PKI identity certificate by the organization's PKI service provider.

The policies associated with this information category are:

- Card Issuance Approval Information can only be entered into IDMS-SCD by a user holding the CardIssue_Approver role (PO-10).
- A person holding CardIssue_Approver role can enter approval information for card applicants belonging to the organizational units (department or division) that have been designated in his/her role indexing parameter (PO-11).
- Any user holding the CardIssue_Approver role cannot be assigned the CardApplicant_Sponsor or Credential_Enroller role (PO-12).
- Apart from entering authorization information (for card production) and unique identifiers (UPN and DN), the CardIssue_Approver is also granted the authorization to generate the card production package. The card production package consists of all information needed for electrical and graphical personalization of a smart ID card for a given holder.

2.4. Physical Facilities Access Identifier information and associated policies

This consists of the full name of the card applicant (potential card holder), a unique identifier associated with the card holder and the expiration date of the card/identifier being sent to the Physical Access Control systems (PACS) located at the facilities where the card holder must be allowed entry.

- The identity credential for a potential card holder can be uploaded to a PACS system at a physical facility from IDMS-SCD only by users holding the PACS_Controller role (PO-13).
- The physical facilities are grouped into regions and a PACS_Controller is assigned one or more regions. Hence a user assigned the PACS_Controller role can only upload information to PACS systems only to those regions specified in his role indexing parameter (PO-14).
- A user holding a PACS_Controller role should not hold any other role in IDMS-SCD (PO-15).
- Since this information includes the name of the card holder (which is a PII) it must be transmitted in a secure manner to all PACS systems from IDMS-SCD (PO-16).

2.5. IT Systems Access Identifier information and associated policies

This consists of the full name of the card applicant and a unique number called UPN being sent to create an Account ID in the corporate directory such as Novell's e-directory or Microsoft's Active Directory. This account entry is used as the user identifier by access control systems in the various IT application systems in the organization or by a centralized authorization server that provides single sign-on capabilities.

- The identity credential for a potential card holder can be uploaded to the corporate directory from IDMS-SCD only by users holding the ITSecurity_Controller role (PO-17).
- A user holding a ITSecurity_Controller role should not hold any other role in IDMS-SCD (PO-18).

2.6. System-level Policies

In addition to policies associated with each information flow, there are some system-level policies that are required to be specified depending upon the type of enterprise environment (overall structure, relative size of various organizational units and geographical dispersion) the smart ID card deployment facility services, as well as the granularity at which privileges are specified in the access related policies.

Based on the enterprise environment, we had to formulate the following system-level policies.

- There should be no more than one sponsor for a given organizational unit (org_unit) (PO-19).
- A credential enroller cannot be assigned more than two regions (PO-20).
- No more than two users can be assigned the ITSecurity_Controller role (PO-21).

Access related policies specify privileges at a coarser level of granularity as they are expressed at the business process level independent of the systems where they will be enforced. However these higher-level privileges must be resolved to transaction-level privileges. Further if roles have indexing parameters, the transaction-level privileges assigned to roles must be further resolved to the level of the values associated with role indexing parameters in order to generate privileges at the user session level. To help these resolutions, certain privilege ordering policies must be specified. The following are examples of this class of policies in the context of our implementation.

- PACS_Controller assigned to a region can upload facilities access identity credentials to all PACS systems in all facilities that come under that region (PO-22).
- The session level privileges for user is the sum of all transactions authorized for all roles assumed by that user in that session, scoped by the role parameter values specified in the user role assignment for that particular user (PO-23).

3. Policy Specification Framework for IDMS-SCD

Having looked at policies for handling different categories of information collected and provisioned for Smart ID card deployment as well as some system-level policies, let us now look at the overall framework that we have developed to represent the data and procedural rules for specification and enforcement of these policies. Our framework consists of the following:

- A structure for representing access specifications (we called this RBAC-SCD)
- A structure for representing different types of policy constraints (SCD_Constraints)
- Encoding of access specification data based on RBAC-SCD (we called it SCD_Access_Data)
- Encoding of policy constraints data based on SCD_Constraints (we gave it the name SCD_Constraints_Data)
- Procedural Rules for validation of access specifications based on policies and instantiation of privileges consistent with policies for performing access mediation (SCD_Policy_Rules).

3.1. Structure for Access Specification (RBAC-SCD)

We used the Role-based Access Control Model (RBAC) [7] as the underlying structure for representing access specifications in the IDMS-SCD scenario as the policies called for the users with specific designated roles for performing the various transactions (data import and data provisioning). We call the instantiation of the RBAC model for this deployment scenario as RBAC-SCD. We used the XML Schema to describe RBAC-SCD. The major building blocks of RBAC-SCD and hence its associated XML Schema elements are:

- Role Definitions (that includes an associated role indexing parameter)

- User-Role Assignments (that includes elements for putting the role indexing parameter name and parameter values for each assignment)
- Role-Privilege Assignments (Generic Privileges)

Parameterized role definition in RBAC-SCD schema:

```
<xs:element name="role"
type="roleType"/>
<xs:complexType name="roleType">
<xs:attribute name="roleID"
type="xs:ID" use="required"/>
<xs:attribute name="rolename"
type="validRole" use="required"/>
<xs:attribute name="role_param"
type="xs:string" use="optional"/>
</xs:complexType>
```

User-Role Assignment:

```
<xs:element
name="UserRoleAssignment"
type="URAType"/>
<xs:complexType name="URAType">
  <xs:sequence>
     <xs:element name="role"
type="xs:IDREF" maxOccurs="1"/>
     <xs:element
name="role_param_value"
type="xs:string"
maxOccurs="unbounded"/>
  </xs:sequence>
<xs:attribute name="user"
type="xs:IDREF" use="required"/>
</xs:complexType>
```

Role-Privilege Assignment:

```
<xs:element
name="RolePrivilegeAssignment"
type="RPAType"/>
<xs:complexType name="RPAType">
   <xs:sequence>
     <xs:element name="privilege"
type="xs:string"
maxOccurs="unbounded"/>
   </xs:sequence>
   <xs:attribute name="role"
type="xs:IDREF" use="required"/>
</xs:complexType>
```

3.2 Structure for Policy Constraints Representation (SCD_Constraints)

Different types of access related policies result in different types of constraints. Addition of new policies may result in new types of constraints. Hence, even though the basic authorization structures (i.e., roles, users and privileges) remain, constraint types may undergo change. Hence for easy of maintenance of the entire policy specification and authorization framework, we decided to develop a separate schema for capturing constraints. Here we reproduce a subset of code from our SCD_Constraints schema to illustrate the type of constraints whose data can be specified in our framework.

The structure for capturing the separation of duty (SSD) policy requires place holders for capturing a pair of roles whose totality of privileges represent conflict of interest.

```
<xs:element name="ssd_roles"
type="SSDType" />
<xs:complexType name="SSDType">
   <xs:attribute name="SSD_ID"
type="xs:ID" use="required" />
   <xs:attribute name="BaseRole"
type="xs:IDREF" use="required" />
   <xs:attribute name="ConflictRole"
type="xs:IDREF"  use="required" />
</xs:complexType>
```

Another example of a constraint is the one that specifies the maximum number of domains (organizational units or regions in our scenario) that can be assigned to a given role in order to limit the scope of privileges.

```
<xs:element
name="Limit_Role_Regions"
type="Limit_Role_Regions_Type" />
<xs:complexType
name="Limit_Role_Regions_Type">
    <xs:attribute name="role1"
type="xs:ID" use="required" />
    <xs:attribute name="max_regions"
type="xs:int" use="required" />
</xs:complexType>>
```

3.3 Access Specification Data based on RBAC-SCD (SCD_Access_Data)

The access specification data based on RBAC_SCD schema for IDMS-SCD domain consists of various roles assigned to perform the transactions in IDMS-SCD, the indexing parameter associated with each role and the separation of duty relationship (conflicting role) each role has with other roles. This access specification data is given in table 1 below:

The XML encoding of the role definition for one of the roles shown in Table 1 (e.g., CardApplicant_Sponsor role), the user-role assignments for this role showing the set of organizational units to which the user is authorized through the role indexing parameter as well as the role-privilege assignments showing the generic privilege are given below:

CardApplicant_Sponsor role specification:

```
<role roleID="CAS"
rolename="CardApplicant_Sponsor"
Role_Param="Org_Unit" />
```

User-Role Assignment specification involving CardApplicant_Sponsor role along with role indexing parameter values assigned to a particular user (organizational units he is authorized to sponsor):

```
<UserRoleAssignment user="SmithJ">
<role>CAS</role>
<role_param_value>Sales
</role_param_value>
<role_param_value>Marketing
</role_param_value>
</UserRoleAssignment>
```

Role-Privilege Assignment data for CardApplicant_Sponsor role is:

```
<RolePrivilegeAssignment role="CAS">
      <privilege>Create New
Applicant</privilege>
      <privilege>Update Applicant
Data</privilege>
</RolePrivilegeAssignment>
```

Table 1 – Roles and Privileges in IDMS-SCD

Role Name	Privileges	Conflicting Roles	Indexing Parameter
CardApplicant_Sponsor	Upload Sponsorship Information Package to IDMS	Credential_Enroller CardIssue_Approver	Organizational Unit
Credential_Enroller	Upload Enrollment Information Package to IDMS	CardApplicant_Sponsor CardIssue_Approver	Region
CardIssue_Approver	1. Record Approval for Card Production after assigning unique UPN and DN data for Card Applicant. 2. Provisions Card Production Package to Card Management Systems 3. Updates Statuses for the issued cards (Card Lifecycle Management)	CardApplicant_Sponsor, Credential_Enroller	Organizational Unit
PACS_Controller	Provision PACS Data for authorized regions	CardIssue_Approver	Region
IT_Security_Controller	Provision Account Data for Enterprise Directory	CardIssue_Approver	NONE

3.4. Policy Constraints Data based on SCD_Constraints (SCD_Constraints_Data)

The policy constraints data based on SCD_Constraints schema provides information for instantiating the generic policies in terms of the domain data pertaining to roles, users, privileges etc. The data pertaining to the constraint that a CRE role (Credential_Enroller) cannot be assigned to more than 2 regions whose structure was shown in section 3.2 is encoded as:

<Limit_Role_Regions role1="CRE"
max_regions="2" />

3.5. Policy Rules for Access Specification Validation and Privilege Resolution (SCD_Policy_Constraints)

The access specification data and policy constraints data both encoded in XML completes the declarative specification of policies for IDMS-SCD domain. For the procedural specification, we developed XSLT transforms since the declarative specification data is encoded in XML. The XSLT transforms were used in two ways to complete the policy specification framework.

- Apply the policy constraints data on the access specification data to validate whether access specifications do conform to the policy constraints requirements.
- Resolve the generic privileges stated in the access specification to transaction-level privileges and scope the transaction-level privileges based on the role parameter values assigned to a given user to generate a user's session-level privileges.

Example of a Policy Validation Rule:The XSLT transform that validates whether user role assignments do not violate the role parameter values limit specified for a given role is as follows:

```
<xsl:comment>
Constraint 3: Limit # of regions for
a role.
</xsl:comment>
<xsl:for-each
select="$constraints/Model_Constrain
ts/Limit_Role_Regions">
  <xsl:variable name="role1"
    select="@role1"></xsl:variable>
  <xsl:variable
    name="max_regions1"
    select="@max_regions">
  </xsl:variable>
<xsl:for-each
select="$data/RBAC_SCD/UserRoleAssig
nment">
  <xsl:variable name="user1"
    select="@user"></xsl:variable>
<xsl:for-each select="RoleItem[role
= $role1]">
  <xsl:variable name="ParamCount"
select="count(region)">
  </xsl:variable>
<xsl:if test="$ParamCount >
$max_regions1">
Constraint 3 Violation ------------
--------------------
User <xsl:value-of select="$user1"
/> with role
<xsl:value-of select="$role1" /> is
assigned to
<xsl:value-of select="$ParamCount"
/> regions.The maximum number of
regions allowed is
<xsl:value-of select="$max_regions1"
/>.
</xsl:if>
</xsl:for-each>
</xsl:for-each>
</xsl:for-each>
```

The outcome of the application of the XSLT transform on the access specification data using the model constraint data that a Credential_Enroller cannot be assigned more than 2 regions results in the following violation identification:

```
      Constraint 3 Violation -----
      User SteveQ with role
      CRE is assigned to
      3 regions.
      The maximum number of regions
allowed is 2.
```

Example of a Privilege Resolution Rule:The following XSLT transform generates session-level privileges for a user, by consolidating all the role assignments for the user and the indexing parameters for those role assignments as well as the privileges assigned for those roles.

```
<xsl:comment>
      Generate session-level
privileges.
</xsl:comment>
============================
Generating session-level
privileges...
<xsl:for-each
select="$data/RBAC_SCD/UserRoleAssig
nment">
   <xsl:variable name="user1"
    select="@user">
   </xsl:variable>
<xsl:for-each select="RoleItem">
    <xsl:variable name="role1"
select="role"></xsl:variable>
    <xsl:variable name="ou1"
select="ou"></xsl:variable>
    <xsl:variable name="region1"
select="region"></xsl:variable>
----------------------------
User: <xsl:value-of select="$user1"
/>
Role: <xsl:value-of select="$role1"
/>
OU: <xsl:value-of select="$ou1" />
Region: <xsl:value-of
select="$region1" />
<xsl:for-each
select="$data/RBAC_SCD/role_privs[@r
ole = $role1]">
   <xsl:variable name="privs1"
select="privilege"></xsl:variable>
Privileges: <xsl:value-of
select="$privs1" />
</xsl:for-each>
</xsl:for-each>
</xsl:for-each>
```

The application of the above XSLT transform generates the following session-level privileges for a user:

```
----------------------------
User: VincentH
Role: CAS
OU:
Region:
Privileges:
CREATE_NEW_APPLICANT
UPDATE_APPLICANT REMOVE_APPLICANT
```

4. Benefits and Summary

A number of XML vocabularies exists for policy specification in web environments [8,9] as well as for access specification [10]. But we chose to develop our customized XML Schema based on standardized RBAC model [7] for representing the structure of access specification, XML for encoding access specification data and use of XSLT for validating access specifications for conformance to policies as well as for privilege resolution, because of the simplicity and flexibility the overall framework provided us. The use of a common encoding scheme removes possibility of any errors introduced due to incorrect semantic mappings in policy implementations when multiple representations are used for different elements of policy representation and enforcement. Further, the platform-neutral representation makes specification and enforcement of access rights in all systems that interact with IDMS-SCD easier. The above features thus provide the potential for application of the framework described in this paper to any large scale infrastructure systems made up of heterogeneous components in domains such as banking transactions and supply chain.

The primary contribution of this paper is the use of XSLT transforms for procedural realization of all policies that govern the multiple data flows involved in the backend infrastructure used for smart ID card deployment instead of proprietary and complex policy specification languages. These transforms together with an RBAC model with extended features provides a sophisticated access control framework for an environment consisting of heterogeneous systems handling privacy-sensitive data. The dynamic generation of session-level privileges enables the development of a reference monitor that provides access mediation without a great deal of rule processing overhead.

- Ease of security administration without common bottlenecks such as role and privilege proliferation
- Preservation of employee privacy and realization of the added security due to smart card-based authentications for physical facilities and IT system access.

5. References

[1] R.Chandramouli, "Infrastructure Standards for Smart ID Card Deployment", IEEE Security & Privacy Magazine, Volume 5, Issue 2, March-April 2007. pp. 92 – 96.

[2] Health Insurance Portability and Accountability Act of 1996. Available at http://frwebgate.access.gpo.gov/

[3] Gramm-Leach-Bliley Act Requirements at http://www.ftc.gov/bcp/conline/pubs/buspubs/glbshort.shtm

[4] M. Backes, G. Karjoth, W. Bagga, and M. Schunter. Efficient comparison of enterprise privacy policies. In *Proceedings of the 2004 ACM Symposium on Applied Computing*, pages 375–382. ACM Press, 2004.

[5] M. Backes, B. Pfitzmann, and M. Schunter. A toolkit for managing enterprise privacy policies. In *European Symposium on Research in Computer Security (ESORICS)*, volume 2808 of *LNCS*, pages 101–119. Springer–Verlag, 2003.

[6] A. Barth and J. C. Mitchell. Enterprise privacy promises and enforcement. In *WITS '05: Proceedings of the 2005 Workshop on Issues in the Theory of Security*, pages 58–66, New York, NY, USA, 2005. ACM Press.

[7] D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli. Proposed NIST standard for role-based access control. *ACM Transactions on Information and System Security*, 4(3):224–274, 2001.

[8] A. Anderson, A. Nadalin, B. Parducci, D. Engovatov, E. Coyne, F. Siebenlist, H. Lockhart, M. McIntosh, M. Kudo, P. Humenn, R. Jacobson, S. Proctor, S. Godik, S. Anderson, and T. Moses. Extensible access control markup language (XACML) version 2.0, 2004.

[9] Security Assertion Markup Language (SAML) v2.0,http://www.oasis-open.org/specs/index.php#samlv2.0

[10] B.Shafiq, E.Bertino and A.Ghafoor. Access Control Management in a Distributed Environment Supporting Dynamic Collaboration. In *Proceedings of the 2005 workshop on Digital identity management, Nov 2005.*