Access Specification and Validation Framework for Secure Smart ID Cards Deployment

Ramaswamy Chandramouli
National Institute of Standards and Technolog
Gaithersburg, MD 20899, USA (mouli@nist.gov),

and

Stephen Quirolgico
National Institute of Standards and Technology
Gaithersburg, MD 20899, USA (stephen.Quirolgico@nist.gov)

ABSTRACT

Deployment of smart cards as identity tokens (Smart ID Cards) requires the support of an enterprise system called Identity Management System (IDMS) for collection, storage, processing and distribution of personal identity credentials. Secure configuration of IDMS for Smart ID Card deployment (IDMS-SCD) requires an access specification and validation framework that is platform-neutral and derives the security requirements based on detailed business processes analysis and application of robust security principles. In this paper, we describe the development and implementation of such a framework. The access and policy specifications in this framework are represented using XML Schema and XML and the validation of the access specification for conformance to policies is performed using XSLT.

Keywords: Identity Management, Smart Cards, Access Control, XML, XSLT

1. INTRODUCTION

To enable secure access to physical facilities and IT systems, the U.S. Government issued a directive called HSPD-12 calling for "reliable and tamper proof credentials". As part of this directive, the National Institute of Standards and Technology (NIST) developed a government-wide standard [1] that lays out the process requirements connected with collection, storage and dissemination of various forms of identity credentials (biographic, biometric, organizational) from employees and contractors of U.S. Government. The standard also calls for secure storage of these credentials (or a subset there of) on a secure smart card and holders of these smart cards are to be authenticated using one or more of some prescribed authentication modes (or use cases) depending upon the sensitivity-level of the physical facility or the criticality-level of the IT system for which the card holder is seeking access.

To comply with this directive, all agencies of U.S Government have to develop an infrastructure for deployment of smart cards (which we will call as Smart ID cards) for personal identity verification. This deployment calls for two major types of Tasks. They are:

- IDV- Task 1: Central Management of Identity credentials of all agency affiliates (employees and contractors). We call this task the Credential Lifecycle Management.
- IDV-Task 2: Electronic Verification of Smart ID Card-based Credentials.

In this paper our focus is on IDV-Task 1 (secure and centralized management of personal identity credentials). This task is facilitated by a class of enterprise IT system called the Identity Management System (IDMS). Though the functional features of the commercial IDMS offerings vary widely, there is consensus within the community that the two canonical functions of any IDMS are: (a) Provide suitable data stores for collection, storage and processing of various types of credentials and (b) Provide a workflow engine that will support the tasks of importing credentials from multiple sources and provisioning of those credentials to various identity-enabled applications based on a pre-defined sequence.

Hence the overall security of the government's Smart ID Card deployment depends upon the secure configuration of IDMS-SCD (IDMS version for Smart Card Deployment) system since the latter performs the key task of credential lifecycle management. The foundation for a secure IDMS-SCD configuration is a robust access control (authorization) specification and validation framework. The main contribution of this paper is the development of such a framework.

The salient features of our access specification and validation framework are:

- It is platform neutral (it can be used for secure configuration of any IDMS used for Smart ID card deployment
- The deployment-specific security policies (expressed in the form of policy rules) are derived from a detailed analysis of the business processes supported by IDMS-SCD by applying a set of security principles to those processes.
- It is self-validating (it contains the tools necessary to validate the access specifications for conformance to a set of deployment-specific security policies). Thus there is complete traceability of each access specification to the underlying policy.

The overall organization of this paper is as follows. The building blocks for our access specification and validation framework are described in Section 2. The development methodology for our access specification and validation framework and the implementation aspects are covered through sections 3 through 5. The benefits of our framework are summarized in chapter 6.

2. ACCESS SPECIFICATION & POLICY VALIDATION FRAMEWORK – BUILDING BLOCKS

The development and Implementation of the Access Specification and Policy Validation Framework has the following steps:

- AS-PVF – Step 1: Develop the Security Requirements for IDV-Task 1 (Credential Lifecycle Management) through the following approach:
  (a) Detailed Analysis of the business processes involved in the Task
  (b) Application of Security Principles to the various processes involved in the Task to derive the overall security requirements.
  (c) Derivation of Security Policies to address the security requirements
- AS-PVF – Step 2: Tabulate the Access and Policy Rule Specification Data. The columns of this table represent the data attributes that are a consequence of the Security Principles while the rows represent an access specification that also contains within it data for expressing the policy rules.
- AS-PVF – Step 3: The implementation of the underlying access control model for access specification using XML Schema. The additional structures based on the model entities that are needed for capturing policy data are also developed using XML Schema.
- AS-PVF – Step 4: Encoding of the Access Specification and Policy Specification Data in XML based on the associated Schemas.
- AS-PVF – Step 5: Encoding of the Validation Logic (which contains the encoding of policy rules embedded in it) for verification of access specification data for conformance to policy rules (instantiated using policy data) using XSLT.

The derivation of security requirements for credential lifecycle management (IDV-Task 1) is described in sections 3.1. The policy rules for addressing the requirements are derived in section 3.2. Section 4 provides the complete tabulation of access and policy specification data (to meet the security requirements) along with illustration of the traceability of the data to the security principles. The structures needed for representing the chosen access control Model and the policy models and their associated XML encoded data are the focus of section 5. This section also contains the validation logic code - for validating the access specification data for conformance to policy rules.

3. SECURITY REQUIREMENTS & POLICY RULES FOR SMART ID-CARD DEPLOYMENT

The nature of data and the process dynamics dictate the applicable security principles and the latter in turn is used to derive the security requirements. Part of the requirements can be met through access specification and part has be to realized through specification and enforcement of policy rules.

3. 1 Derivation of Security Requirements

The methodology for deriving security requirements for credential lifecycle management using IDMS-SCD consists of:

- Analyzing the dynamics of the individual processes that constitute the IDV-Task 1 in terms of the data flows.
- Applying Security Principles to the Data Flows based on data content and the role of the particular data flow in the overall credential lifecycle management for deriving the security requirements
- Deriving Policy Rules for addressing the security requirements

Based on FIPS 201 [1] standard, the credential lifecycle management processes identified are:

- Card Sponsorship
- Credential Enrollment
- Credential Approval
- Card Producer
- Card Issuance/Activation
- Credential Provisioning to Physical Access control (PACS) systems
- Credential Provisioning to Logical Access Control (LACS) systems

Card Sponsorship involves a responsible official of the enterprise sending the biographic and organizational affiliation information for a potential, eligible smart card holder whose identity needs to be electronically verified. Credential Enrollment consists of an officer of the enterprise or an authorized contractor performing the identity proofing of the sponsored applicant using some breeder documents (e.g., Birth Certificate) and collecting

biometric data such as fingerprint minutiae or digital facial image and sending them to IDMS-SCD.
Credential Approval involves a high-ranking security officer authorizing the issue of Smart Card to the applicant. Card Production consists generating the graphical (visual) and electronic credential sets respectively for printing and electronic personalization of a smart card to be issued to the applicant. The card issuance/activation is the process of generating artifacts called digital signatures attesting the credentials and also providing artifacts such as Public Key Infrastructure (PKI) digital certificates attesting the legitimacy of the credential issuer which also contains data (e.g., the public key) for verifying the attestation. The credential provisioning to PACS systems involves upload of relevant credentials to door access panels for controlling entry of smart card holders to various facilities such as buildings, computer centers etc. The process of provisioning to LACS systems involves upload of identity credentials to centralized authorization repositories for authorizing IT resources (e.g., data and/or application programs) such as Enterprise Directories or native access control repositories such as those found in domain controllers etc.

An analysis of the business processes described so far reveals that all of them with the exception of credential approval involve generation of data flows either into the IDMS-SCD or out of it [2]. The overarching security issues associated with these data flows is that they involve privacy-sensitive data and also the end-product of these processes is an artifact (i.e., Smart Card) that provides the right of passage to access physical facilities and IT systems of the enterprise where it is deployed. These security issues were the drivers for Security Principles (though this mapping process cannot be fully described in a paper of this size). The security principles thus arrived at, were applied to the process analysis to arrive at overall security requirements for the credential lifecycle management Task and are listed below. Please note the security principle based on which the requirement was formulated is given in parenthesis.

- IDVT1-SR1: IDMS-SCD processes must be carried out only by appropriate role holders (Proper Authorization)
- IDVT1-SR2: IDMS-SCD roles being generic must be parameterized to facilitate assignment of only as many privileges as needed for a role holder consistent with his/her allocation of duties based either on an organizational division or geographic region. (Principle of Least Privilege)
- IDVT1-SR3: In order to maintain the overall integrity of IDMS-SCD operations, certain combination of processes should not be authorized for the same individual and this can be enforced by designating pairs of roles as conflicting so that they are not assigned to one individual. (Static Separation of Duty)
- IDVT1-SR4: Certain pairs of IDMS-SCD processes cannot be performed in a single user session. This can be achieved by preventing users from activating the associated roles in a single session (Dynamic Separation of Duty)
- IDVT1-SR5: Escalation of privileges in IDMS-SCD should be restricted by restricting the number of role holders who hold identical privileges (Privilege Containment)
- IDVT1-SR6: IDMS-SCD should enforce context-based authorization (or role activation) since there is a specific sequence for its processes (enforced through a workflow engine)

3.2 Addressing Security Requirements through Policy Rules

Not all security requirements in the set (IDVT1-SR1 through IDVT1-SR6) can be met through static assignments. Security Policy Rules have to be defined and enforced for meeting many requirements at various stages of security configuration of IDMS-SCD. These stages span security administrator operations as well as the development of policy enforcement point in the IDMS-SCD security reference monitor. Some of these stages with respect to the driving security principles are: (a) During Role Assignment (Separation of Duty) (b) During Role Activation in a session (Dynamic Separation of Duty) and (c) During Role-Privilege binding in a session for parameterized roles (Privilege Containment). An example of security policy rules for the enrollment data flow operations is given below:

- A person holding 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 (IDVT1-SR2).
- A person holding the Enroller role cannot be assigned the Sponsor, Identity_Management_Officer (IMO) or Card_Producer roles (IDVT1-SR3)
- A user assigned to both enroller and card_activator roles cannot activate both roles in the same user session (IDVT1-SR4)
- An instantiation of an enroller role with permissions for a particular region cannot be assigned to more than 2 users (IDVT-SR5)

4. TABULATION OF ACCESS AND POLICY SPECIFICATION DATA

The application of security principles on the individual IDMS-SCD process operations has yielded us the security requirements and the policy rules to realize those security requirements. The next step is the tabulation of all access specification data as well as all the parameters associated with them so as to ensure that all security requirements are captured in the IDMS-SCD static security configuration and the encoded policy rules can be properly instantiated for enforcement during IDMS-SCD run time. A way to obtain this assurance is that the tabulated data items collectively provide coverage for all security principles that were the drivers for the security requirements and eventually for policy rules as well. The tabulation of access specification data along with their parameters is given in Table 1.

Table 1: Access and Policy Specification Data

IDV-Task1 Process	Authorized Role	Indexing Parameter	Conflicting Role	Assignment Cardinality Limit
Card Sponsorship	Sponsor	Organizational Unit (OU)	Enroller, IMO, Card_Activator	One for each OU
Credential Enrollment	Enroller	Region	Sponsor, Card_Activator (for same session only)	Two for Each Region
Credential Approval	IMO	NONE	PSO, ISO, Card_Producer	One
Card Production	Card_Producer	NONE	IMO	Two
Card Issuance/Activation	Card_Activator	Region	Sponsor, Enroller (for same session)	Two for Each Region
Credential Provisioning to PACS	PSO	Facility	IMO	Two for Each Facility
Credential Provisioning to LACS	ISO	IT System Class	IMO	Two for Each IT System Class

The coverage analysis of the above data for all the driving security principles that yielded the security requirements (section 3.1) and policy rules (section 3.2) is given below in Table 2:

Table 2: Coverage Analysis of Tabulated Access & Policy Specification Data

Security Principle	Column Name
Proper Authorization	Assigned Role
Principle of Least Privilege	Indexing Parameter
Static Separation of Duty	Conflicting Role
Dynamic Separation of Duty	Conflicting Role
Privilege Containment	Assignment Cardinality Limit

5. IMPLEMENTATION OF ACCESS SPECIFICATION AND VALIDATION FRAMEWORK

Access Specification is structured representation of authorization data based on an underlying access control model. We chose the Role-based Access Control (RBAC) [3,4] since it is standardized, provides administrative ease, capable of supporting different types of policy rules and widely implemented in many commercial products. For the overall implementation of our Access Specification and Validation framework, we used the following artifacts:

- Structure for Access Control Model (AC-MODL) and specification of access control data (AC-DATA)
- Structure for Policy Rules Elements (PO-MODL) and specification of policy Rules data (PO-DATA)
- Rules or Validation Constraints that specify the conditions for conformance of access specification to policy rules instantiated using policy rules data (PO-VERF)

5.1. Representation of Access Control Model (AC-MODL) and Access Control Data (AC-DATA)

We used XML Schema [5] to describe the structure of the RBAC model that we used for access configuration of IDMS-SCD. XML Schema language constructs were used for User & Role definitions, User-Role Assignments and Role-Privilege Assignments. Sample XML Schema definitions for User-Role Assignments and Role-Privilege Assignments are given below:

```
<xs:element    name="UserRoleAssignment"
type="URAType" />
<xs:complexType name="URAType">
    <xs:sequence>
      <xs:element  maxOccurs="unbounded"
ref="RoleItem" />
```

```
    </xs:sequence>
    <xs:attribute          name="user"
type="xs:IDREF" use="required" />
  </xs:complexType>
```

Access specification encoded in XML that corresponds to the above Schema are given below:

```
<UserRoleAssignment user="VincentH">
 <RoleItem>
    <role>ENR</role>
    <region>EAST</region>
   <region>NORTHEAST</region>
 </RoleItem>
 </UserRoleAssignment>
```

5.2 Representation Policy Rules Elements (PO-MODL) and Policy Rules Data (PO-DATA)

To carry policy rules data, it is necessary to develop an underlying policy model. We used XML Schema to represent this policy model. An example of the policy model structure used for carrying data about the pair of conflicting roles (so as to enforce separation of duty) is given below followed by an example of the associated Policy data encoded in XML.

```
<xs:complexType
name="Separation_of_Duty_Type">
    <xs:attribute      name="base_role"
type="xs:ID" use="required" />
    <xs:attribute  name="conflict_role"
type="xs:ID" use="required" />
  </xs:complexType>
<Model_Constraints
xmlns:xsi="SCD_Constraints.xsd">
       <Separation_of_Duty
base_role="SPN" conflict_role="ENR" />
       <Limit_Role_Param_Value
role="SPN" param="SALES"
       param_values_limit="1" />
       <Limit_Role_Param_Value
role="ENR" param="EAST"
       param_values_limit="2" />
       <Limit_Role_Regions    role="ENR"
max_regions="2" />
</Model_Constraints>
```

5.3 Representation of Validation Constraints for Policy Conformance Verification (PO-VERF)

We now have the access and policy specification data for IDMS-SCD. The last step in the implementation of the overall access specification and validation framework for IDMS-SCD is the procedural application of instantiated policies (instantiated using policy rules data (PO-DATA)) on the access specification data (AC-DATA) to verify whether the latter conforms to the required policy rules. For this purpose, we developed XSLT transforms [6] that contains the policy conformance logic operating on the XML files containing the access specification data and policy rules data.

The XSLT transform that validates whether user role assignments do not violate the role parameter values limit specified for a given role (to limit the span of privileges for a role holder to enforce least privilege) is as follows:

```
<xsl:comment>
Constraint 3: Limit # of regions for a
role.
</xsl:comment>
<xsl:for-each
select="$constraints/Model_Constraints/
Limit_Role_Regions">
<xsl:variable              name="role1"
select="@role1"></xsl:variable>
<xsl:variable name="max_regions1"
             select="@max_regions">
<xsl:variable>
<xsl:for-each
select="$data/RBAC_SCD/UserRoleAssignme
nt">
<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 policy rule data that an Enroller cannot be assigned more than 2 regions results in the following violation identification:

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

6. BENEFITS AND SUMMARY

We presented a development methodology and implementation approach for a comprehensive access & policy specification and validation framework that can be used for secure configuration of an infrastructure system

(i.e., IDMS-SCD in our case) used for Smart ID card deployment. The Secure configuration has as its basis, a complete set of security requirements associated with the main task of IDMS-SCD (i.e., Credential Lifecycle Management). The security requirements in turn are derived by applying robust security principles that are appropriate for the sensitivity of data and integrity of the underlying processes in the credential lifecycle management. It is this methodology that distinguishes our framework from some of the other related approaches such as [7]. The innovative aspects of our framework are the in-depth process analysis, choice of appropriate security principles, the proper application of these principles to derive security requirements, an in-built validation element that tests the access specifications for conformance to policy rules and the last but not the least the use of platform-neutral implementation using XML Schema, XML and XSLT. The last feature facilitates its widespread use as it enables configuration of any commercial IDMS product that an agency may choose to deploy for its Smart ID card deployment.

7. REFERENCES

[1]http://csrc.nist.gov/publications/**fips**/**fips201-1**/**FIPS-201-1**-chng1.pdf, March 2006.

[2] R.Chandramouli, Philip Lee, "Infrastructure Standards for Smart ID Card Deployment", **IEEE Security & Privacy Magazine**, Volume 5, Issue 2, Mar-Apr 2007. pp. 92 – 96.

[3] 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 Securit***y*, 4(3):224–274, 2001.

[4] D. F. Ferraiolo,, D. R. Kuhn, and R. Chandramouli. Role-based Access Control. *Artech Book House,* Jan 2005.

[5]. XML Schema Part 0: Primer W3C Recommendation, May 2001, http://www.w3.org/TR/xmlschema-0/

[6] XSLT: http://www.w3.org/TR/xslt

[7] J.B.D Joshi et al. "Access Control Language for Multidomain Environments", **IEEE Internet Computing**, Nov-Dec 2004.