

# An XML-Based Security Management Model for Multilevel Security Databases

Awad Awadelkarim and Norbik Idris

Faculty of Computer Science and Information System, University Technology Malaysia, Malaysia

**Abstract:** *This paper proposes a universal architecture of Security Management Model (SMM) for integrating security features namely low-level access control at row and column level of relational legacy databases using XML as an integration medium. The proposed model integrates and then rearranges, controls, and manages the new and inherent low-level access control attributes based on Rule-Based Algorithm (RBA) and Global Security Policies (GSP) of the integrated environment. In addition, the paper shows how Public-Key Cryptography Scheme (PKCS) can be adapted by the SMM to provide a consistent security management at the global level.*

**Keywords:** *Database security, integration of databases, legacy databases, access control, PKCS, XML.*

*Received September 25 2004; accepted February 23, 2005*

## 1. Introduction

Integration of heterogeneous legacy databases is important, as information is often required to be synthesized and aggregated especially for high-level management. When security issues are taken into consideration, such integration presents several interesting problems. Firstly, the legacy databases are assumed heterogeneous and as such have incompatible security features. Queries that may involve data from the legacy databases need to be managed by different security management systems. In addition, there are specified security features for each database schema, and integration may require a new post-integration security management model.

In this paper, we propose an XML-based Security Management Model (SMM) for integrating security features of multilevel security relational legacy databases, and then managing the integrated security features. The proposed model integrates and then rearranges, controls, and manages the new and inherent low-level access control attributes based on Rule-Based Algorithm (RBA) and Global Security Policies (GSP). The SMM offers two types of security management namely Default Security Management (DSM) and Exception-handling Security Management (ESM) to fulfill such environment requirements. Furthermore, recently, a great deal of interest has been expressed in implementing and extending Public-Key Cryptography Scheme (PKCS) into standard authentication protocols and distributed systems [10, 11, 19, 20, 23]. Thus, the paper shows also how PKCS can be adapted by our SMM to provide a consistent security management in the integrated environment. With PKCS adaptation, the SMM prevents the

vulnerabilities that can be caused by inconsistency of security features at the global level.

The organization of this paper is as follows. Section 2 presents detailed description for the SMM architecture, SMM components, and the PKCS adaptation. In section 3, we present the related work. Section 4 provides a conclusion of this paper.

## 2. Related Work

Numerous research has addressed the problem of security conflicts. To our best knowledge none has addressed the conflict or contradiction of low-level access control of integrated heterogeneous legacy relational databases using XML. Thus, our SMM perceives the inconsistency of the security features due to heterogeneity during the integration process, and salvages that at the global level. Furthermore, the global policy assumed to be well defined, well verified, and implemented in simple and secure manner. More to the point, lately there has been a considerable interest in environment that support multiple and complex access control policies and use of XML as integration medium [2, 3, 4, 5, 21]. Previous work has mostly been tackled within the frame of federated databases with a global schema that secluded by access control and restrictions [8]. Different proposals have been addressed various problems in this framework such as [1, 4, 6, 8, 9, 12, 13, 14]. Alternatively, there has been considerable research interest in language-based approaches to access control [3, 7, 15, 16, 17, 18, 22] and the main goal is to provide a language that can support multiple access-control policies and achieve separation of policies from mechanisms.

Therefore, original contributions of the proposed SMM are: We propose two types of integration process namely physical and logical to ensure optimal-extensive security features integration. In addition, the SMM also offers two sorts of security management: Default and exception-handling security managements to fulfill the environment requirements. Moreover, we extend the security-control management by adapting PKCS with DSM to provide authentication, and with ESM to offer both authentication and secrecy to prevent the vulnerabilities that can be caused by inconsistency of the security features. Furthermore, our SMM deals with and supports Mandatory Access Control (MAC). Simultaneously, the SMM is designed to include features that can support and satisfy Discretionary Access Control (DAC) requirements. Additionally, the SMM architecture is able to make use of Role-Based Access Control (RBAC) model to comply with all its associated functions and specifications, which make the SMM easy to be adopted and customized by an industry particularly with using of the meta-integration-language XML. Altogether reflect the strength and show the distinction of our proposed model with indicated related work.

### 3. SMM Architecture

When the various databases are required to integrate, heterogeneity becomes the core and focal concern. With the several variety of heterogeneity, our SMM handles the security heterogeneity that is low-level security features to ensure secured and homogeneous access at the global level. The SMM architecture consists of five basic components namely Objects' Security Features Integration (OSFI) unit, Integrated Security Features Management (ISFM) unit, Rule-Based Algorithm (RBA), Global Security Policies (GSP), and XML Repository.

Thus, the three main players in the SMM are Subjects (users), Objects (row or column of data) and Labels (security-sensitivity labels). Thus, the SMM controls and manages the objects access in three dimensions: Object-Label denotes and specifies the sensitivity of the object and determines the criteria that must be met for a subject to access that object. Subject-Label denotes the label authorization assigned to a subject. Exception-Label denotes an exception label that can be given to a particular subject to access certain object, which is beyond the subject's label-authorization in an exceptional session.

Various industries use different sensitivity-label schemes to implement low-level access control. Table 1 illustrates a variety of typical label scheme applied by industry and shows the short numeric form used by our SMM to indicate the exact sensitivity-label. Thus, higher numbers indicate more sensitive and lower numbers indicate less sensitive. The same sequential numbers used for subject sensitivity labels. This

sequential numbers meets the SMM design and implementation requirements, and at the same time, it can be easily adopted and then customized by an industry.

Table 1. Typical label scheme.

Industry	Sensitivity Label	Short Numeric Form Used by the SMM
Defense/ Military	Top Secret	4
	Secret	3
	Confidential	2
	Unclassified	1
Financial Services	Acquisition	4
	Corporate	3
	Client	2
	Operations	1
Judicial	National Security	3
	Sensitive	2
	Public	1
Health Care	Primary_Physician	3
	Patient_Confidential	2
	Patient_Release	1
Business to Business	Trade_Secret	4
	Proprietary	3
	Company_Confidential	2
	Public	1
HR and other Systems	Highly_Sensitive	4
	Sensitive	3
	Confidential	2
	Public	1

Eventually, the scenarios like so, numerous subjects have different level of authorizations (labels/clearance) request to access objects that are labeled according to their security-sensitivity. Thus, the SMM is concerned in integrating and then managing the security features without taking into account the technique used to enforce low-level access control at each local database side and its implementation mechanism as well.

### 3.1. The XMLRepository

#### 3.1.1. Security Centric Document

Security Centric Document (SCD) contains all the security information about the subjects and objects at the global level. In fact, during the integration process, the SMM maps all information related to the low-level security features that provided by each local database side to the equivalent XML data-centric-documents as shown in Figures 1 and 2. Then, SMM integrates the all mapped-centric documents based on RBA and GSP in one comprehensive (global) structured document that is SCD as shown in Figure 3.

#### 3.1.2. Object Security Centric Document

Object Security Centric Document (OSCD) is XML structured data-centric-document that derived from pervious global SCD based on RBA and GSP. It contains only the integrated security features of the objects to determine the objects that required to be integrated, level of integration (physical/logical), and

the new (global) objects’ sensitivity labels as shown in Figure 4.

```
<databaseA>
  <rowobj>
    <objid> ... </objid>
    <objname> ... </objname>
    <objtype> ... </objtype>
    <objdesc> ... </objdesc>
    <objsenlab> ... </objsenlab>
    <objlink> ... </objlink>
    <objrem> ... </objrem>
  </rowobj>
  ...
</databaseA>
```

Figure 1. Object security information (XML Data-Centric –Doc.).

<p><b>Object Security Information:</b>  <b>Obj_id:</b> Object identification indicates database, table, and row/column that the object belongs to.  <b>Obj_name:</b> Stand for object name.  <b>Obj_type:</b> Object type (char, integer, and so forth).  <b>Obj_desc:</b> Object description (describe the object and its semantic).  <b>Obj_sen_label:</b> Object’s sensitivity label (values: 1, 2, 3, or 4).  <b>Obj_link:</b> The link between the object and the others (optional).  <b>Rem:</b> General remark about the object (optional).</p>
--

```
<databaseA>
  <rowsubj>
    <subjid> ... </subjid>
    <subjtag> ... </subjtag>
    <subjname> ... </subjname>
    <subjdesc> ... </subjdesc>
    <subjsenlab> ... </subjsenlab>
    <subjrem> ... </subjrem>
  </rowsubj>
  ...
</databaseA>
```

Figure 2. Subject security information (XML Data-Centric-Doc.).

<p><b>Subject Security Information:</b>  <b>Subj_id:</b> Subject identification.  <b>Subj_tag:</b> Subject tag denotes local database side that it belongs to.  <b>Subj_name:</b> Stand for subject name.  <b>Subj_desc:</b> Subject description (position/ranking) (optional).  <b>Subj_sen_label:</b> Subject’s sensitivity label (values: 1, 2, 3, or 4).  <b>Rem:</b> General remark about the subject (optional).</p>
--

### 3.1.3. Label-Permit Centric Document

The SMM generates a Label-Permit (certification) to be utilized for upgrading process of the subject sensitivity label. Thus, the SMM maps the related label permit information that provided by each local database side to the equivalent XML data-centric-document. Then, the SMM integrates all the mapped centric documents based on RBA and GSP in one comprehensive (global) document that is Label-Permit Centric Document (LPCD) as shown in Figure 5.

### 3.1.4. Garbage Centric Document

In the SMM, when neither physical nor logical integration are applicable for certain object, the integration is denied, and the object will be temporary

sent to the Garbage Centric Document (GCD) to be excluded from that particular integration session (performance efficiency), however, after the integration session is completed, all the objects in the GCD will be retained to the SCD.

```
<scdglobal>
  <databaseA>
    <objtab>
      <row>
        <objid> ... </objid>
        <objname> ... </objname>
        <objtype> ... </objtype>
        <objdesc> ... </objdesc>
        <objsenlab> ... </objsenlab>
        <objlink> ... </objlink>
        <objrem> ... </objrem>
      </row>
      ...
    </objtab>
    <subtab>
      <row>
        <subjid> ... </subjid>
        <subjtag> ... </subjtag>
        <subjname> ... </subjname>
        <subjdesc> ... </subjdesc>
        <subjsenlab> ... </subjsenlab>
        <subjrem> ... </subjrem>
      </row>
      ...
    </subtab>
  </databaseA>
  <databaseB>
    ...
  </databaseB>
  ...
</scdglobal>
```

Figure 3. The Security Centric Document (SCD) schematic.

```
<oscdglobal>
  <row>
    <gobjid> ... </gobjid>
    <gobjlevelintg> ... </gobjlevelintg>
    <gobjsenlab> ... </gobjsenlab>
    <gobjtag> ... </gobjtag>
    <gobjrem> ... </gobjrem>
  </row>
  ...
</oscdglobal>
```

Figure 4. The OSCD schematic.

<p><b>Objects’ Security Centric Document:</b>  <b>Gobjid:</b> Global object ID indicates the original integrated objects.  <b>Gobjlevelintg:</b> Global-objects’-level of integration (values: 0 for physical integration and 1 for logical integration).  <b>Gobjsenlab:</b> Global objects’ sensitivity label.  <b>Gobjtag:</b> Global-objects’ tag.  <b>Gobjrem:</b> General remark about the object (optional).</p>
---

## 3.2. Objects’ Security Features Integration Unit

The main function of Objects’ Security Features Integration (OSFI) unit is to integrate the objects’

security features. Actually, It determines all the information needed for the integration process, and then generates the global OSCD. The SMM offers two types of integration to ensure coherent integration. Physical based on the objects' syntax, names, and types, when the logical based on the objects' semantic.

```

<globalLPCD>
  <databaseA>
    <rowside>
      <sideid> ... </sideid>
      <upglevel> ... </upglevel>
      <mintaglab> ... </mintaglab>
      <rem> ... </objrem>
    </rowside>
  ...
</databaseA>
<databaseB>
  ...
</databaseB>
...
</globalLPCD>

```

Figure 5. The global LPCD schematic.

<p><b>Label-Permit Information:</b>  <b>Side_id:</b> Side identification indicates the destination-database side.  <b>Upgrade_level_allow:</b> Indicates number of upgrading levels allowed (0 = No level, 1 = one level, or 2 = two levels).  <b>Min_target_label:</b> Specifies the minimum allowable sensitivity-label for upgrading at that destination side (can be 1, 2, or 3).  <b>Rem:</b> General remark (optional).</p>
---

### 3.2.1. Physical and Logical Integration

In physical integration, PIOSF SMM compares the object name and type with the same attributes of the other objects found in the global SCD. If at least one of the two attributes is matched, then it refers to object description attribute to authenticate the semantic of the objects. After the semantic authentication is confirmed, the integration process is established. In the physical integration, There are two levels: Identical physical integration that when the two above-mentioned attributes are matched, and one-degree physical integration with one attribute matching.

When the physical integration is not applicable, we propose an alternative technique to be implemented that is logical integration. Logical integration is mainly based on objects' descriptive attributes. LIOSF evaluates the semantic among the objects with reference to the objects' remark-text-values, which are located in the SCD. Once the matching process is approved, the integration process can be executed. Otherwise, the integration process is denied, and the particular object will be sent to the GCD.

### 3.2.2. Objects' Security Labels Integration Processor

We classify the integration process based on the number of objects that are prepared for integration into two procedures: Two-Objects Integration Process

(TOIP) that when only two objects are required to integrate, and Cluster-Objects Integration Process (COIP) with group (more than two) of objects.

For the two selected objects, TOIP acquires the associated sensitivity-label values from the global SCD. Then, TOIP evaluates the two values based on RBA and GSP. Thus, if they are equal, the same value will be assigned to the integrated-objects'-label, which is located in the global OSCD. If not, TOIP computes the difference between the two values (i. e., the subtraction-value), and then applies the absolute rule to that value. If the end-result-value equals to 1, TOIP picks the higher label-value between the two values, and then applies the upgrading policy. Subsequently, the higher chosen label-value will be assigned to the integrated-objects'-label. Then, tagging policy is applied. Otherwise, the integration of the two objects is denied. Also, to maintain the security of the objects at the global level and to ensure optimal -secure integrated environment, our SMM forbids the integration of the objects with difference is not equal to 1.

For group of objects, COIP obtains the associated objects' security labels-values from the global SCD. Then, COIP compares the values, and if they are equal, the same value will be assigned to the global objects' label, which is located in the global OSCD, If not, COIP arranges the specific objects in a set of *n* distinct objects, (*n* indicates number of objects). Then, COIP performs *r-Permutation and Combination* of the set of *n* distinct objects. When *r* denotes the element ordering selected from *n* distinct objects. In our model *r*-value is constant, *r* = 2. So, the *2-combination* of a set of *n* distinct objects is:

$$C(n, r) = P(n, r) / r!$$

$$P(n, r) = n! / (n - r)! \text{ thus } C(n, r) = n! / (n - r)! * r!$$

$$\text{since } r = 2 \text{ thus } C(n, 2) = n! / (n - 2)! * 2!$$

When the subsets of the *C(n, 2)* are built, COIP computes the difference (the subtraction value) between the two values of each subset, and then applies absolute rule for that value. Then, COIP stores the end-result-value of each subset in an array, and the index (subscript) of the array denotes the two objects that construct the subset. COIP examines the array, and all subsets with cell-value = 0 or 1 will be integrated. COIP chooses the highest value among the selected subsets to be assigned to the objects' integrated label at the global level in the OSCD. Then the tagging policy is applied. Otherwise, when the cell-value is not equal to 0 or 1, COIP divert the particular subset to be handled by the TOIP. Eventually, the ultimate outputs of this processor will be sorted in the global OSCD.

### 3.3. Integrated Security Features Management

The main function of this unit Integrated Security Features Management (ISFM) is to manage the integrated security features at the global level. In fact,

ISFM handles the users' queries and requests to control the objects' accessibility based on RBA and GSP.

As mentioned earlier, our SMM offers two types of security management (DSM and ESM) to provide consistent low-level access control. In DSM, the default treatment presents the subjects' fixed-security labels obtained during the integration process. DSM-labels apply to all received queries excluding those with exception-handling requests. The fixed-security labels can be revised and updated periodically. DSM only allows access to objects if a subject has a qualified (authorized) sensitivity label. If not, DSM either forwards the query to be handled by the ESM or denies the access. In ESM, the exception treatment presents the temporary security label given to a subject for particular session or special-query handling upon a request and its approval. The ESM-labels can temporarily override the DSM-labels for a particular session to ensure comprehensive accessibility (Access Rule Consistency) and flexible security management at the global level.

### 3.4. PKCS Adaptation

As indicated earlier, our SMM adapts the PKCS to prevent the vulnerabilities that can be caused by inconsistency of security features at the global level. Simultaneously, PKCS adaptation ensures dependable and consistent security management in the integrated environment. PKCS is adapted in our model to provide authentication with DSM, and to offer both authentication and secrecy with ESM. In PKCS, each party generates a pair of keys namely Public Key (KU) and Private Key (KR). The two keys use for the encryption and decryption of messages. Each party publishes his public key in public-accessible site and keeps his private key secret (private). Generally, public-key algorithms (e. g., RSA and Elliptic Curve) rely on one key KU for encryption and a different but related key KR for decryption [19]. However, either of the two related keys can be used for encryption, with other used for decryption. This enables a rather different cryptographic scheme to be implemented (i. e., Secrecy, Authentication, or both). Therefore, as an initial step, each subject in our model is required to generate a pair of keys and then submit the public key to PKCS\_Admin through a Submission Process (SP).

#### 3.4.1. Submission Process

In the Submission Process (SP), each subject submits an authorized Submission-Request (SR) that has been signed using the private key of the local database-side security administrator (LocalDB\_Admin). The SR consists of Subject Identification and the Public Key of the Subject. The PKCS\_Admin validates the received SR by using the public key of the localBD\_Admin. Consequently, the subject's public key will be

endorsed and authorized. Later, the PKCS\_Admin uses the authorized public key to authenticate the messages (verify the signature), which are received from the associated subject. In addition, PKCS\_Admin encrypts the outgoing messages (secrecy) using the authorized public key of the subject (the recipient). In any case, only the public keys that are published by the PKCS\_Admin (with the PKCS\_Admin's Signature) are certified.

$$SR = E_{KR_{LocalDB\_Admin}} [Subj\_id, KUs]$$

#### 3.4.2. PKCS with DSM

In DSM, to authenticate the queries that received from the subject, the subject is required to attach a Query-Request (QR) with each query. In fact, our SMM automatically generates a QR for each subject's query. The QR consists of Subject Identification, Subject Default Sensitivity Label, and Query-request's Status (Default or Exception). Then, DSM requests the subject to sign (encrypt) the QR using the private key of the subject to produce a digital signature as shown in Figure 6. On the other hand, DSM verifies the signature by decrypting the QR using the public key of the subject that provided by PKCS\_Admin. If the signature is valid the associated query will be moved forward.

$$QR = E_{KR_s} [Subj\_id, Subj\_def\_sen\_lab, Query\_stat]$$

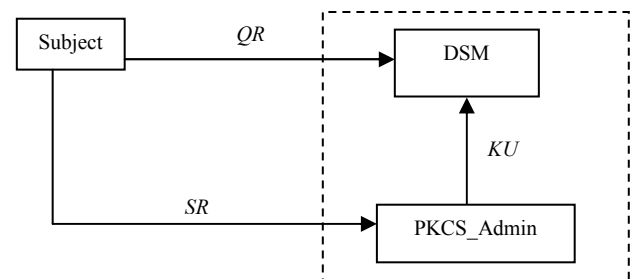


Figure 6. DSM: Subject query request process.

#### 3.4.3. PKCS with ESM

The same authentication process made previously by DSM will be implemented by ESM as well (Figure 7). Moreover, in ESM, the subject has Non-qualified (illegitimate) sensitivity-label to access the exact objects. Therefore, ESM authenticates the subject's tag and the objects' tags from the SCD and the OSCD respectively. If they are marked with the same tag, immediately the access is denied by ESM. If not, ESM requests the subject to provide a Label-Permit (LP) for that exceptional query.

Therefore, ESM generates a Label-Request (LR) for that query and asks the subject to sign. At the same time ESM attaches the authorized public key of the associated subject with another copy of the same LR, and then signs the total LR using the private key of the

PKCS\_Admin, as in (1). Next, the signed LR will be sent to the LPCD\_Admin. On the other hand, LR is encrypted using the subject's private key and then sent to LPCD-Admin. The LR consists of Subject Identification, Object Identification(s) that required for access, and the Request Time Stamp (i. e., time this request was issued), as in (1) and (2). Actually, the subject submits the signed LR to LPCD-Admin requesting for a Label Permit (LP). Therefore, LPCD-Admin decrypts the two LR's that received from the ESM and the Subject using the public key of the PKCS\_Admin and the public key of the associated subject respectively. In fact, LPCD-Admin decrypts the LR that received from the ESM to obtain the authorized public key of the subject. Then, LPCD-Admin uses the legitimate subject's public key to verify and authenticate the second LR's originator. If the authentication is valid and the two decrypted LR's are matched, LPCD-Admin refers to the SCD, OSCD, and LPCD, which placed in the XML repository, to validate and confirm the credentials-ability of upgrading the subject-sensitivity label. If the upgrading request is approved, LPCD\_Admin issues the Label Permit to the particular subject.

The LP consists of two parts: Confidential part and Subject part. The confidential part includes Subject Identification, Upgraded (new) Sensitivity Label, Object Identification(s) allowable for access, LP's Time Interval (lifetime of this LP, and this time to prevent replay after LP has expired), as in (3). To validate the authorized party that issues the LPs, the confidential part will be signed using the private key of LPCD\_Admin (to confirm this LP issued from LPCD\_Admin). In addition, LPCD\_Admin encrypts the signed confidential-part using the public key of the PKCS\_Admin to protect the integrity of the data (i. e., to ensure no one, except the PKCS\_Admin, can read or alter the information even the subject that LP is belong to), as in (3). The Subject Part consists of the Confidential-part plus the Subject Identification, Subject Default Sensitivity Label, and the Request Status (e. g., approved or rejected), as in (4). LPCD\_Admin signs the subject part using the public key of the subject to ensure only the subject can identify and read the subject's LP among the others. Then, LPCD\_Admin sends the complete LP, as in (5), to the exact subject. Later, the subject presents the received LP to the ESM to access the required-exact objects after a verification process using the private key of the PKCS\_Admin, and then the public of the LPCD\_Admin. Otherwise, the access is denied by the ESM. Conclusively, Figure 8 demonstrates the complete SMM architecture.

$$LRI = E_{KR_{PKCS\_Admin}} [[Subj\_id, Obj\_id(s), Req\_time\_stamp] + KUs] \tag{1}$$

$$LR_2 = E_{KR_s} [Subj\_id, Obj\_id(s), Req\_time\_stamp] \tag{2}$$

$$LP\_Confidential\_Part (LP\_CP) =$$

$$E_{KU_{PKCS\_Admin}} [E_{KR_{LPCD\_Admin}} [Subj\_id, Subj\_upg\_sen\_lab, Obj\_id(s), LP\_time\_stamp]] \tag{3}$$

$$LP\_Subject\_Part (LP\_SP) =$$

$$E_{KU_s} [[Subj\_id, Subj\_def\_sen\_lab, Req\_stat] + LP\_Confidential\_Part] \tag{4}$$

$$Thus: LP = E_{KU_s} [[Subj\_id, Subj\_def\_sen\_lab, Req\_stat] + [E_{KU_{PKCS\_Admin}} [E_{KR_{LPCD\_Admin}} [Subj\_id, Subj\_upg\_sen\_lab, Obj\_id(s), LP\_time\_stamp]]]] \tag{5}$$

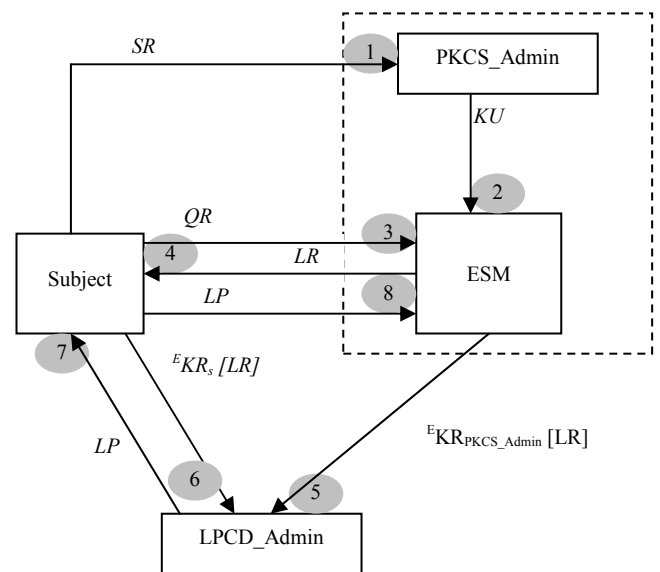


Figure 7. ESM: Subject query request and label permit process, where numbered arrows illustrate the secrecy and authentication stages.

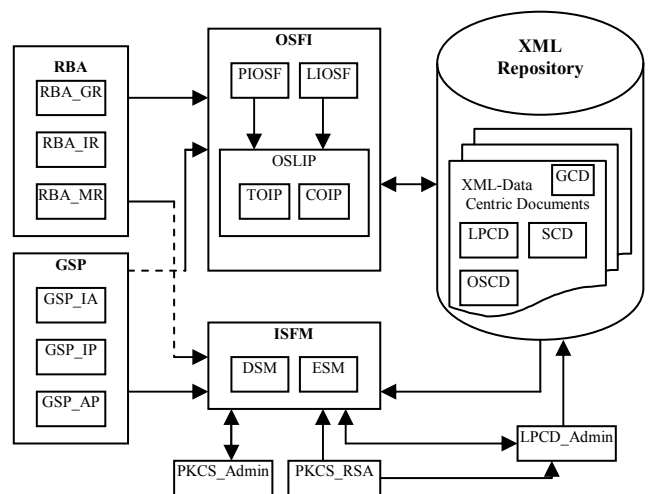


Figure 8. The SMM architecture.

### 3.5. Rule-Based Algorithm

Rule-Based Algorithm (RBA) consists of a collection of procedures and rules that maintain the security and operability of the SMM. The procedures define and

control the operation-flow and relationships between the SMM components. RBA provides general, integration, and management Rules.

### 3.5.1. General Rules

- *GRI*: Each local database side should provide complete security-information about objects, subjects, and subjects' certification approved and signed.
- *GR2*: Subjects' Public-keys should be submitted to the PKCS\_Admin through authorized Submission Process.

### 3.5.2. Integration Rules

RBA\_IRs monitor, control, and verify all events done by OSFI in term of sequence and functioning to integrate the objects' security features.

- *IR1*: Objects Information obtained in GR should be properly mapped to the equivalent XML documents.
- *IR2*: All mapped XML documents must be accurately integrated in one global SCD.
- *IR3*: Objects are required for integration should be precisely determined and the numbers of the objects (TOIP or COIP) should be confirmed as well.
- *IR4*: Integration level (P\LIOSF) should be decided.
- *IR5*: The new (global) sensitivity label should be exactly determined by OSLIP.
- *IR6*: Thus, the objects' security features must be well integrated and stored in the global OSCD.

### 3.5.3. Management Rules

RBA\_MR monitor, control, and verify all events done by ISFM in term of sequence and functioning to manage the integrated security features and handle the users' queries and requests.

- *MRI*: Subjects information obtained in GR should be properly mapped to the equivalent XML documents.
- *MR2*: All mapped XML documents must be correctly integrated in one global LPCD and SCD.
- *MR3*: Query type (Read or Write) and query status (Default or Exceptional) should be well classified.
- *MR4*: Default subject's sensitivity label, object's (s') sensitivity label(s), and all associated tags must be obtained (SCD+OSCD) and accurately evaluated.
- *MR5*: Thus, the query-management is decided (DSM or ESM) and then properly handled.

## 3.6. Global Security Policies

Global Security Policies (GSP) consists of a collection of policies designed to support the RBA to standardize and homogenize the objects-accessibility and its procedures at the global level. On top, GSP for determining authorization as a basis for the access-

control decision that made by our SMM to achieve the desired security level. In addition, GSP offers a universal and flexible/adaptable platform to implement the SMM in various industries. GSP offers Integration and Access/Management Policies.

### 3.6.1. Integration Policies

During the Integration Process (IP) the following policies are implemented: *IP1: Labeling Policies*:

- *IP1\_A*: Objects with equal sensitivity labels: The same label will be assigned to the integrated (global) object.
- *IP1\_B*: Objects with different sensitivity labels: *IF* the difference = 1 *then* the higher label will be assigned to the integrated (global) object. *Else* (i. e., the difference  $\diamond > 1$ ) the integration is denied.
- *IP2: Upgrading Policy*: Only one upgrading-level is allowed, when difference between the sensitivity-labels-values = 1. In fact, upgrading values (0 = No level allowed, 1 = one level, or 2 = two levels) (Default value is 1). However, *IP2\_A* only allows one-level-upgrade to maintain the security.
- *IP3: Tagging Policy*: Each object and subject must be tagged to indicate which database-side it belongs to.
- *IP4: Denying Policy*: *If* ((the objects' label- values are not equals) *AND* (their difference  $\diamond > 1$ )) *then* the integration of security features is denied.
- *IP5: Garbage Policy*: Object must be temporary sent to the GCD, when neither physical nor logical integration are applicable.

### 3.6.2. Access Policies

During the query-handling process and the integrated security features management the following policies are implemented:

- *AP1*: Subjects must have Qualified Sensitivity Labels to access the exact objects (i. e.,  $\text{subject\_label} \geq \text{object\_label}$ ).
- *AP2*: ESM forbids all subjects' queries for Exception-handling when ( $\text{subject\_label} = 1$  or 4).
- *AP3*: ESM prohibits all subjects' queries for Exception -handling when ( $\text{objects\_labels} > \text{subject\_label}$  *AND*  $\text{subject\_tag} = \text{object\_tag}$ ).
- *AP4*: ESM permits subjects' requests for Exception-handling to be proceeded when ( $\text{subject\_label} = 2$  or 3 *AND*  $\text{subject\_tag} \diamond \text{object\_tag}$ ).
- *AP5: Upgrading Policy*: Only subject with labels = 2 or 3 is allowed to be upgraded to 3 and 4 respectively. This to specify the minimum subject's sensitivity-label allowed for upgrading at a particular destination side. So, values can be 1, 2, or 3. However, *AP6* only allows 2 or 3 because

subject's label = 1 is not allowed to maintain the security, and 4 is non-upgradeable.

#### 4. Conclusion

In this paper, we have proposed a universal structure of security management model for integrating low-level access control security features of heterogeneous legacy databases using XML. The Model handles the integration and management of the security features at the global level to ensure consistent and secure access. The model is composed of five major components namely OSFI, ISFM, RBA, GSP, and XML Repository.

In the SMM, we have proposed two types of integration process namely physical and logical to ensure optimal-comprehensive integration. In addition, we have also proposed two sorts of security management: Default and Exception-handling Security Managements (D/ESM) to fulfill such environment requirements. Moreover, this paper has demonstrated how PKCS can be adapted by the SMM to prevent the vulnerabilities that can be caused by inconsistency of the security features. Thus, we have extended the security-control management by adapting PKCS with DSM to provide authentication, and with ESM to offer both authentication and secrecy.

Currently, we are working on implementing XML-Java-based prototype of the SMM. The implementation will be validated and precisely tested and evaluated. However, a validation test has been conducted during the earlier analysis and design stages. In addition, we plan to extend our SMM in other research to handle the security management of the Web services document as demand increase for Internet substance and its related application such as digital libraries, e-learning, medicine and so forth. On the other hand, we plan also to expand the SMM to cover other security aspects of integrated databases such as data integrity, confidentiality, and authentication.

As mentioned earlier, our SMM deals with and supports Mandatory Access Control (MAC) to enforce security in the integrated environment and we adapt PKCS just to ensure consistent security management. PKCS Adaptation within our SMM enhances the security management in the integrated environment by extending the low-level access control to grant authentication and secrecy with the DSM and ESM. Consequently, the benefit of PKCS Adaptation is improving the security and scalability throughout the SMM framework as shown in Figure 8 to comply with the environment requirements. In addition, the achievement of PKCS security fully trusts in the private keys secrecy, and the authorized availability of the associated public keys. Thus, this achievement is addressed by the SMM with PKCS adaptation through the submission process (section 3), which handles the generation, submission, and distribution of the

authorized public keys amongst the SMM components. In addition, our SMM design enhances the PKCS adaptation and its performance by collaborating the administrations of local database sides and the global administration to offer such commitments. However, in relation to symmetric cryptography scheme, obviously, the inherent PKCS performance shortcomings concerning the computational requirements and key size that can decelerate the performance, it cannot be prevented, but it can be minimized. In our case, when only a few amount of information needs to be exchanged and authenticated, that distinction is insignificant and negligible. In addition, as indicated earlier, the SMM structure adapts PKCS in way that not only minimizes the shortcomings, but also avoids using any key distribution or management system, and collaborating the administrations of local database sides and the global administration to handle the adaptation, so the total overheads can be significantly reduced. Moreover, beyond the secrecy and authentication, and as shown in section 3 with ESM, our PKCS adaptation methodology can offer additional security services such as Data Integrity and Non-Repudiation.

#### References

- [1] Agrawal R., Evfimievski A., and Srikant R., "Information Sharing Across Private Databases," in *Proceedings of ACM SIGMOD*, San Diego, CA, 2003.
- [2] Bertino E. and Catania B., "Integrating XML and Databases," *IEEE Internet Computing Journal*, vol. 5, pp. 85-88. 2001.
- [3] Bertino E., Buccafurri F., Ferrari E., and Rullo R., "A Logical Framework for Reasoning on Data Access Control Policies," in *Proceedings of the 12<sup>th</sup> IEEE Computer Security Foundations Workshop*, Alamitos, CA, pp. 175-189, 1999.
- [4] Bhatti R., Bertino E., Ghafoor A., and Joshi J., "XML-Based Specification for Web Services Document Security," *IEEE Computer Society Journal*, pp. 41-49, April 2004.
- [5] Bird P., "Implementing Low Level Access Control with DB2UDB," *The IDUG Solutions Journal*, vol. 7, 2000.
- [6] Chandramouli R., "Business Process Driven Framework for Defining an Access Control Service Based on Roles and Rules," *Research Report*, Computer Security Division, ITL NIST, Gaithersburg, 2000.
- [7] Cholvy L. and Cuppens F., "Analyzing Consistency of Security Policies," in *Proceedings of the IEEE Symposium on Security and Privacy*, CA, pp. 103-112, 1997.
- [8] Dawson S., Qian S., and Samarati P., "Providing Security and Interoperation of Heterogeneous

- Systems,” *Research Report*, Kluwer Academic Publishers, Boston, pp. 1-29, 1999.
- [9] Dawson S., Qian S., and Samarati P., “Secure Interoperation of Heterogeneous Systems,” in *Proceedings of the IFIP 14<sup>th</sup> International Conference of Information Security*, Budapest, Vienna, 1998.
- [10] Downard I., “Public-Key Cryptography Extensions into Kerberos,” *IEEE Journal*, vol. 21, no. 5, pp. 30-34, 2002.
- [11] Erdem O. M., “High-speed ECC Based Kerberos Authentication Protocol for Wireless Applications,” in *Proceedings of IEEE Global Telecommunications*, vol. 3, pp. 1440-1444, 2003.
- [12] Fillingham D., “Exploration of the Use of Partition RBAC for Medical Applications,” *Research Report*, US Department of Defense, 1998.
- [13] Gardarin G., Mensch A., Tuyet T., and Smit L., “Integrating Heterogeneous Data Sources with XML and Xquery,” *Research Report*, e-XMLMedia, Bourg La Reine, France, 2001.
- [14] He Q., “Privacy Enforcement with an Extended RBAC Model,” *Research Report*, North Carolina State University, USA, 2003.
- [15] Jajodia S., Samarati P., and Subrahmanian V. S., “A Logical Language for Expressing Authorizations,” in *Proceedings of the IEEE Symposium on Security and Privacy*, CA, pp. 31-42, 1997.
- [16] Jajodia S., Samarati P., Sapino L. and Subrahmanian V., “Flexible Support for Multiple Access Control Policies,” *ACM Transactions on Database Systems*, vol. 26, no. 2, pp. 214-260, 2001.
- [17] Jajodia S., Samarati P., Subrahmanian V. S., and Bertino E., “A Unified Framework for Enforcing Multiple Access Control Policies,” in *Proceedings of the ACM International SIGMOD*, pp. 474-485, 1997.
- [18] Ryutov T. and Neuman, C., “Representation and Evaluation of Security Policies for Distributed System Services,” in *Proceedings of the IEEE DARPA Information Survivability Conference*, CA, 2000.
- [19] Stallings W., *Cryptography and Network Security*, Prentice Hall, New Jersey, 2003.
- [20] Wan T., Ng C., and Poh G., “Integrating Public Key Cryptography into the Simple Network Management Protocol Framework,” in *Proceedings of TENCON*, vol. 3, pp. 271-276, 2000.
- [21] Wang L., Wijesekera D., and Jajodia S., “Towards Secure XML Federations,” *Research Report*, Center of Secure Information Systems, George Mason University, USA, 2003.
- [22] Woo T. and Lam, S., “Designing a Distributed Authorization Service,” in *Proceedings of IEEE INFOCOM*, 1998.
- [23] Yee L. and De Silva L., “Application of Multi-Layer Perceptron Networks in Public Key Cryptography,” in *Proceedings of the International Joint Neural Networks*, vol. 2, pp. 1439-1443, 2002.



**Awad Awadelkarim** received his BSc in computer science and information systems from SUST, his MSc in computer engineering from IIUM. Currently, he is a PhD candidate in computer science at University Technology Malaysia, Malaysia. His research interests include information systems security, information integration, and XML.



**Norbik Idris** is a professor and director for the Centre for Advanced Software Engineering, University Technology Malaysia, Kuala Lumpur. His research interests are in the area of information security and software engineering.