CHAPTER 13

Integrating Multilevel Security Policies in Multilevel Federated Database Systems

Marta Oliva¹ and Fèlix Saltor²

¹*Dept. Informàtica i Enginyeria Industrial, Universitat de Lleida, C. Jaume II, 69, E-25001 Lleida (Catalonia).*

²*Dept. Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Campus Nord-Mòdul C5, Jordi Girona Salgado, 1-3, E-08034 Barcelona (Catalonia).*

Key words: Multilevel security, integration, interoperation, federated DBMS.

Abstract: This paper describes a multilevel security policies integration methodology to endow tightly coupled federated database systems with a multilevel security system. The proposal is based on a schema integration process. It obtains, in a semi-automatic form, the ordered set of classification levels for the multilevel security system of the federation, and the translation functions between each ordered set belonging to each component database and the federated ordered set as well. The proposed methodology also includes a way to solve the problem of classification of the components of the Federated Schema generated during the integration process.

1. INTRODUCTION

The growing need for information sharing, among independent entities which have their own database systems, is often solved by a *Federated DataBase System* (FDBS) ([SL90]). There are different aspects to take into account when building and operating such a FDBS: one of them is data protection.

Although the independent entities, named *Component DataBases* (CDBs), have their own access control mechanisms (based on *Discretionary Access Control* (DAC) or *Mandatory Access Control* (MAC)), accesses to the federated system need take into account not only security features of CDBs, but also security features of the federation ([Per93]). The mechanism used by our FDBS to protect data, and more particularly to allow or forbid access, is *MultiLevel Security* (MLS, [BL75]), because its own operation not only helps to control access but also to control information flow.

A FDBS can be *tightly coupled* or *loosely coupled* ([SL90]). Since we deal with strict access control policies, such as MLS, we assume tightly coupled FDBSs only, with administrators at the federated level. We present a methodology of integration of security policies to be used in building tightly coupled federated systems by MLS-CDBs integration.

The main concepts and notation used in this paper to describe the methodology can be found in section 2. Section 3 presents the principal characteristics of the schema integration methodology, upon which our methodology of integration of MLS systems is based. The core of the paper is in section 4 and contains the development of the proposed methodology. The paper ends with a comparison with other research and conclusions in section 5.

2. MAIN CONCEPTS AND NOTATION

A MLS system assigns a *clearance level* to each subject and a *confidentiality level* to each object. Levels assigned to subjects and objects should belong to either a partial or linear (total) ordered set, or a lattice of classification levels (or labels). At this stage we assume that every system uses a linear ordered set. The natural manner to represent an ordered set is through a *Hasse diagram* ([SM77]). The cardinality of an ordered set is the number of elements that are members of the set of classification levels.

When a FDBS is built it is necessary to take into account that although different preexistent CDBs have a MLS system, it is impossible to presume that their ordered sets of classification levels could coincide neither on the number of levels (cardinality), nor on confidentiality meaning ([MLTS92, DJ94]). That is why it is essential to have a mechanism that helps in obtaining the ordered set of the federation MLS system itself (from now on we will call it *Federated Ordered Set* (FOS)).

As accesses at the federated level are subject to the mechanisms that CDBs have to protect data, the FOS of classification levels must take into account the preexistent ordered sets. So the CDBs integration process has to

make an integration process of ordered sets of classification levels of distinct preexistent MLS systems.

As an integration of ordered sets of classification levels we understand the process needed to deduce the equivalence among classification levels belonging to different ordered sets. It is important to note that it is possible that a level member of a specific ordered set does not coincide with any levels belonging to whichever of the remainder ordered sets to integrate.

Taking into account that classifications at the component level must be maintained at the federated level, the conclusion is that the FOS needs to have at least the same number of classification levels as the ordered set having the biggest number of classification levels of all ordered sets. Moreover, in the worst case it is possible that the FOS has as many classification levels as the sum of all ordered set cardinalities, due to the possibility of lack of any equivalence among classification levels of distinct ordered sets.

Given $(S_i, \leq_i)$, $i = 1, \dots, N$, the ordered sets of the N preexisting CDBs, each S_i represents the set of classification levels of its correspondent CDB, and each $\leq_i$ is a binary relation that defines the order among elements of S_i . Through integration process of ordered sets, we obtain a new ordered set $(S_F, \leq_F)$ (FOS) where S_F is the set of classification levels of the federated MLS system and $\leq_F$ is the binary relation that indicates the order among elements of S_F . The cardinality of S_F ($|S_F|$) is bounded by the cardinalities of S_i ($|S_i|$):

$$|S_k| \leq |S_F| \leq (\sum_{i=1}^N |S_i|) \quad \text{where } |S_k| \geq |S_i| \quad \forall i = 1, \dots, N$$

depending on the quantity of classification levels from an ordered set that coincide with any classification level of another ordered set.

With the help of the integration process the FOS $(S_F, \leq_F)$ is obtained, and also the translation functions $(f_i, i=1, \dots, N)$. A translation function allow us to translate each classification level from an ordered set to the FOS with the preservation of the order from original ordered set:

$$x, y \in S_i \quad x \leq_i y \Leftrightarrow f_i(x) \leq_F f_i(y)$$

3. THE SCHEMA INTEGRATION METHODOLOGY

[GSSC95] describes a schema integration methodology based on BLOOM (BarceLona Object Oriented Model, [CSGS94]) that allows obtaining a *Federated Schema* (according [SL90], [ROSC97]). The BLOOM data model is used because it is a semantically rich model (it has distinct kinds of specializations and aggregations) capable of expressing not only semantics already expressed in local schemas, but also additional semantics

very useful for integration tasks. The schema integration methodology is divided into three phases:

- *Semantic Enrichment*: enriches semantically local schemas of the CDBs, which were expressed by some traditional models. It starts with a step of knowledge acquisition and then local schemas augmented with the knowledge previously discovered are converted into rich Component Schemas expressed in our Canonical Data Model (CDM, [SL90]) BLOOM.
- *Detection*: identifies semantically related objects (that belong to different CDBs), through a comparison process guided by a strategy. The strategy operates at two levels: at the coarse level the strategy identifies pairs of specializations to be compared, taking into account the generalization dimension, and at the fine level the strategy identifies pairs of classes to be compared based in the aggregation dimension. Later, a criterion, based on aggregation dimension too, is used to yield a degree of similarity between the pair of classes.
- *Resolution*: integrates the BLOOM schemas of the CDBs after identification of semantically related objects using a *discriminated generalization*. In this variant of generalization, the description of the superclass inherits upwards the description of its subclasses (it takes the union of their abstractions), and each object of a subclass, seen as a member of the superclass, has a discriminant attribute. The discriminant attribute is used for integration purposes and it takes as value the name of the database where it comes from.

4. THE SECURITY POLICIES INTEGRATION METHODOLOGY

Our security policies integration methodology complements the schema integration methodology presented in the previous section, in such a way to allow us to get:

- the FOS
- the translation functions between ordered sets
- the classification of the Federated Schema, which is a result of integration process, according to the obtained FOS.

Our methodology produces, in a semi-automatic form, a FOS that only needs to be validated by the federated system administrator, to be used as the ordered set of the MLS system of the federation. If the ordered set does not get the administrator's validation, the integration process would generate a new FOS proposal in order to be validated by the administrator. Validation can be total or partial; so from a partial validation new proposals can be

generated until total validation is obtained. This is because when different ordered sets are integrated there are distinct combinations that can be used as FOS.

The analysis and validation of all possible combinations among classification levels of distinct ordered sets would require lots of administrator's effort. To reduce the number of combinations to analyze by the administrator, our integration process takes advantage of the information related to the data classification stored in *Export Schemas* (according to [SL90, ROSC97]), as well as the information obtained by schema integration process.

Although it is also necessary to complement the semantic enrichment phase to reach a complete integration of CDBs, this paper focuses on the two last phases of schema integration methodology. Particularly, the detection phase, properly complemented, yields the FOS and the translation functions (thanks to the validation of the administrator). Besides, after complementing the resolution phase, this will allow classifying the components of the Federated Schema.

4.1 Detection phase

The detection phase of the schema integration methodology (presented in section 3) produces semantic relationships between components of different Export Schemas. Semantic relationships are expressed by *semantic assertions* and there are two kinds:

- a) equivalence assertion (*Equivalence Semantic Relationship, E-SR*): the two classes represent equivalent concepts
- b) specialization assertion (*Specialization Semantic Relationship, S-SR*): one class represents a superconcept of the concept represented by the other class.

Figure 1 introduces a simple example that allows remembering the operation of the schema integration methodology, and then it is used to illustrate our security policies integration methodology.

Figure 1. CDB1 and CDB2

Specifically, figure 1 describes the Export Schemas of CDB1 and CDB2, their ordered sets of classification levels, and also the classification of each component of the schemas as well.

The five semantic assertions obtained by means of the detection phase of schema integration methodology are shown in figure 2.

Figure 2. Obtained assertions by detection phase

The ordered set integration process uses the information of the semantic assertions. So, with the initial information of ordered sets of CDBs to integrate and assertions obtained by detection phase, the proposal of a FOS is originated in order to let administrator validate it.

Taking into account the hypothesis of homogeneous integration (CDBs belong to the same universe of discourse) it is logical to presume that interest in information privacy will be similar. For that:

1. in an E-SR it is assumed that classification levels of classes which are considered equivalent concepts, are equivalent classification levels.
2. in an S-SR it is assumed that the classification level of the superconcept is $\leq$ than the classification level of the concept (according to [ML92, IGQ94] a concept always has to be classified at a level $\geq$ than the superconcept, because concept needs to inherit superconcept characteristics).

For the analysis and representation of all information, needed by the ordered set integration process, a multiple directed graph (digraph ([SM77]) where multiple arcs are allowed between two vertices, *multidigraph*) is used. The multidigraph is built from Hasse diagrams, which define initial ordered sets to integrate, and necessary arcs corresponding to different assertions obtained by detection phase. The source and target of each arc belong to distinct ordered sets. The representation of each semantic assertion is the following:

1. for each E-SR an arc from the classification level of a class to the classification level of the other class is added to the multidigraph; and also the inverse arc.
2. for each S-SR an arc from the classification level of the superconcept to the classification level of the concept is added to the multidigraph.

It is important to note that all assertions are taken into account, although they seem redundant or produce conflicts, without any verification. The reason for this is to ensure the use of the greater quantity of information, because the feasibility study of each arc, originated from semantic assertions, before its addition to the multidigraph, could produce a final multidigraph with less quantity of information. Each arc produces some conflict depending on the other arcs of the multidigraph, so if the multidigraph is not complete the decision whether to add or to remove an arc could affect further inclusions or removals of other arcs.

According to the example introduced in figures 1 and 2, from the Hasse diagrams of ordered sets $(S_1, \leq_1)$, $(S_2, \leq_2)$, and arcs needed to represent assertions (E-SR, People-Person), (E-SR, Unemployed-Unemployed), (E-SR, Student-Student), (S-SR, Employee-Administrative) and (S-SR, Employee-Worker) the multidigraph shown in figure 3 is obtained.

Figure 3. Multidigraph of our example

Once the multidigraph is obtained, it is necessary to analyze it to identify possible incompatible arcs, because of the incorporation of arcs corresponding to all semantic assertions. Two arcs are incompatible if their existence does not permit maintaining the typical characteristics of initial ordered sets. Because an arc can be incompatible with different arcs, and at the same time these arcs can be incompatible with others, it is necessary to use a mechanism to determine which is the more incompatible arc. As a mechanism we use a *penalization* system based on the detection of cycles with more than two participant vertices.

Figures 4 (a) and (b) show the two kinds of cycles that can be detected in a multidigraph. Every arc involved in a cycle has to be penalized. An arc must accumulate as many penalizations as cycles in which it is involved.

Figure 4. (a) Cycle

Figure 4. (b) Crossed cycle

Particularly, in the multidigraph of our example, there are the cycles: L11-L12-L22-L11, L12-L13-L22-L12, L11-L12-L13-L22-L11, L22-L23-L12-L22, and L13-L22-L23-L12-L13.

To obtain the FOS from the multidigraph it is necessary to do the following steps:

1. To remove, iteratively, the more penalized arc from the multidigraph. The elimination of an arc implies the updating of the penalization of other arcs that were affected by the existence of the eliminated arc.
2. To replace cycles that have two participant vertices, belonging to different ordered sets, by only one vertex. Two vertices participate in a cycle if $\exists ((v1, v2) \wedge (v2, v1))$ where $v1 \in S_i, v2 \in S_j, i \neq j$.
3. To convert the multidigraph obtained into an ordered set.

After calculating the corresponding penalizations of distinct arcs of the multidigraph shown in figure 3, we obtain the next penalizations: (L22, L12) has 1 penalization, (L22, L11), (L12, L22), (L23, L12) have 2 penalizations and (L13, L22) has 3 penalizations. The arc which must be deleted is (L13, L22), because it is the arc with larger penalization. When arc (L13, L22) is deleted then the penalizations of others arcs change to the penalizations: (L22, L11), (L23, L12) have 1 penalization and (L12, L22) has 2. This time, the arc more penalized is (L12, L22), and after removing it the remaining arcs do not have any penalization. So, the resultant multidigraph is that is in figure 5.

Figure 5. Resultant multidigraph

Figure 6. Final digraph

After replacing vertices involved in a cycle by only one vertex, the digraph shown in figure 6 is obtained. It is important to note that arc (L23-L12, L24) is not necessary because they are deduced from the final digraph. So, the obtained digraph defines the FOS to be used by MLS security system of the federation (if it is validated by the administrator), where $S_F = \{L21, L22-L11, L23-L12, L13, L24\}$, $|S_F| = 5$ y $\leq_F = \{(L21, L22-L11), (L22-L11, L23-L12), (L23-L12, L13), (L13, L24)\}$.

The translation functions f_1 and f_2 obtained through the application of the methodology are defined as:

$$f_1(L11) = L22-L11, f_1(L12) = L23-L12, f_1(L13) = L13$$

$$f_2(L21) = L21, f_2(L22) = L22-L11, f_2(L23) = L23-L12, f_2(L24) = L24$$

To finalize the CDBs integration process it is necessary to solve the classification of the components of the integrated schemas, taking into account the classifications of the components of the initial schemas and the FOS of classification levels as well as the translation functions obtained. This resolution of classifications has to be carried out by the complemented resolution phase of the schema integration methodology.

4.2 Resolution phase

The resolution phase of the schema integration methodology gets the Federated Schema after the integration of the Export Schemas of the CDBs. By use of the discriminant generalization similar classes are integrated whenever specializations to which they belong are similar too.

Figure 7. Federated Schema

Taking up again the example in figures 1 and 2, after the resolution phase, the resultant Federated Schema corresponds with the schema shown in figure 7. The class `Employee_db2` has been created as a conformation of CDB1 and CDB2. At the moment, in this schema only the classification of the components of the initial schemas appears.

To classify the Federated Schema taking into account the FOS, it is necessary to perform the following steps:

1. To update the original classifications of Federated Schema components, which already appear in Export Schemas, applying the translation functions obtained in the detection phase.
2. To classify components that were originated by the use of the discriminant generalization, taking into account these conditions:
 - a) if the classification levels of all subclasses are the same then the superclass is classified at the same level of the subclasses.
 - b) if subclasses are classified at different levels then the superclass is classified at the least confidential level where subclasses are classified.
 - c) it is important to note, as a special property of the discriminant generalization, that when a superclass inherits upwards the union of the abstractions of all its subclasses, a multilevel classified superclass is obtained, although classes of CDBs were not multilevel classified. If an abstraction only appears in a subclass then its original classification level is maintained (with the corresponding translation). Otherwise, an abstraction that appears in several subclasses will be upward inherited depending on its semantics. If classification levels of the subclass abstractions are distinct then abstractions are semantically different, so they will be upward inherited as different abstractions (maintaining its classification).

Figure 8 shows the final result of resolution phase of our security policies integration methodology, where it is possible to see the Federated Schema totally classified according to the FOS.

Although our integration process takes into account only two ordered sets, the integration of a large number of initial ordered sets is done by repeating the same integration process in pairs. The strategy consists in using a ladder integration sequence starting with two ordered sets of the highest cardinality, or two of the larger cardinality, and later, the process is repeated using the FOD obtained from the previous process and one of the larger cardinality among remaining ordered sets ([OS00]).

According to [GQ96], a federated/interoperable system has to comply with the following principles:

1. Principle of autonomy: any access permitted within an individual system must be also permitted under secure interoperation.
2. Principle of security: any access not permitted within an individual system must be also denied under secure interoperation.

Let's see, by an example, how both principles are fulfilled if FDBS uses the classified Federated Schema shown in figure 8. Remembering another time the CDBs introduced in figure 1, a user of the CDB1 having the clearance level L11 can only access class People. If the same user, having the same clearance level, was a user of the federation, after applying the corresponding translation function the equivalent clearance level obtained is level L22-L11. So, through the federated system this user can also only access the class People (from CDB1). Besides, the same user can access the information of the federation related to Fpeople_person, Femployee, Funemployed, Person, Unemployed db2, Employee db2, and Worker.

5. COMPARISON WITH OTHER RESEARCH AND CONCLUSIONS

Our security policies integration process methodology allows us to integrate, in a semi-automatic form, different CDBs taking into account data protection. This data protection aspect is only slightly studied in the federated system field. The process of the methodology presented in this paper integrates CDBs that have MLS as security systems, but the main difference with the proposal presented in [IGQ94] is that the integration is performed although ordered sets of classification levels of different systems do not coincide. Besides, as in [IGQ94], the proposed methodology complements a schema integration methodology, so it can offer a very important support to the administrator of a federated system; contrary to the proposals presented in [GQ96, BSS96] where an administrator, or somebody else, has to establish mappings between distinct ordered sets.

References

- [BSS96] P.A. Bonatti, M.L. Sapino and V.S. Subrahmanian. Merging Heterogeneous Security Orderings. In E. Bertino, G. Kurth, H. Martella and E. Montolivo, editors, *Computer Security - ESORICS 96 (4th European Symposium on Research in Computer Security, Rome, Italy, September 25-27, 1996, Proceedings)*, volume 1146 of LNCS, pages 183-197, Springer-Verlag, 1996.
- [BL75] D.E. Bell and L.J. LaPadula. Secure computer systems: Unified exposition and multics interpretation. Technical Report MTR-2997, (AY/W 020 445), The MITRE Corporation, Bedford, MA, Jul 1975.
- [CSGS94] M. Castellanos, F. Saltor and M. García-Solaco: A Canonical Data Model for the Interoperability among Object-Oriented and Relational Databases. In Özsu, Dayal and Valduriez (eds), *Distributed Object Management*, pages 309-314, Morgan Kaufmann, 1994.
- [DJ94] K.R. Dittrich and D. Jonscher. Current Trends in Database Technology and Their Impact on Security Concepts. In J. Biskup, M. Mongersten and C.E. Landwehr (eds), *Database Security VIII (A-60)*, Elsevier Science B.V. (North Holland) IFIP, pages 11-33, 1994.
- [GQ96] L. Gong and X. Qian. Computational Issues in Secure Interoperation. *IEEE Transactions on Software Engineering*, 22(1):43-51, January 1996.
- [GSSC95] M. García-Solaco, F. Saltor and M. Castellanos. A Structure Based Schema Integration Methodology. In *Proc. 11th Int. Conference on Data Engineering*, Taipei. IEEE-CS Press, 1995.
- [IGQ94] N.B. Idris, W.A. Gray and M.A. Qutaishat. Integration of Secrecy Features in a Federated Database Environment. In T.F. Keefe and C.E. Landwehr, editors, *Database Security VII (A-47)*, pages 89-109. Elsevier Science B.V. (North-Holland) IFIP, 1994.
- [ML92] J.K. Millen and T.F. Lunt. Security for Object-Oriented Database Systems. In *Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy*, Oakland, California, pages 260-272, May, 1992.
- [MLTS92] M. Morgenstern, T. Lunt, B. Thuraishingham and D. Spooner. Security issues in federated database systems: panel contributions. In C.E. Landwehr and S. Jajodia, editors,

- Database Security V(A-6): Status and Prospects*, pages 131-148. Elsevier Science B.V. (North Holland) IFIP, 1992.
- [OS00] M. Oliva & F. Saltor. Integrating Multilevel Security Policies in Multilevel Federated Database Systems. In *Proc. 14th Annual IFIP WG 11.3 Working Conference on Database Security*, Schoorl, The Netherlands, August 21-23, 2000.
- [Per93] G. Pernul. Canonical Security Modeling for Federated Databases. In D.K. Hsiao, E.J. Neuhold, and R. Sacks-Davis, editors, *Interoperable Database Systems (DS-5)(A-25)*, pages 207-222. Elsevier Science Publishers B.V. (North-holland) IFIP, 1993
- [ROSC97] M.E. Rodríguez, M. Oliva, F. Saltor and B. Campderrich. On Schema and Functional Architectures for Multilevel Secure and Multiuser Model Federated DB Systems. In S. Conrad, W. Hasselbring, A. Heuer, G. Saake, editors, *Proceedings of the International CAiSE'97 Workshop on Engineering Federated Database Systems (EFDBS'97, Barcelona)*, Otto-von-Guericke-Universität Magdeburg, Fakultät für Informatik, preprint Nr. 6, pages 93-104, 1997.
- [SL90] A.P. Sheth and J.A. Larson. Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases. *ACM Computing Surveys*, 22(3): 183-236, September 1990.
- [SM77] D.F. Stanat and D.F. McAllister. *Discrete Mathematics in Computer Science*. Prentice-Hall International Editions, 1977.