ARES 2008

The Third International Conference on Availability, Security and Reliability

PROCEEDINGS

March 4-7, 2008 Barcelona, Spain

Edited by Stefan Jakoubi, Simon Tjoa, and Edgar R. Weippl

Organised by

[SECURE] Business Austria

DEXA

In cooperation with

UNIVERSITAT POLITÈCNICA DE CATALUNYA

enisa European Network and Information Security Agency

OESTERREICHISCHE COMPUTER GESELLSCHAFT AUSTRIAN COMPUTER SOCIETY

Proceedings of the

The Third International Conference on Availability, Security, and Reliability

March 4-7, 2008, Barcelona, Spain

CPS

Los Alamitos, California

Washington • Tokyo

IEEE computer society

IEEE Computer Society Order Number P3102
ISBN 0-7695-3102-4
ISBN 978-0-7695-3102-1
Library of Congress Number 2007909935

Additional copies may be ordered from:

IEEE Computer Society
Customer Service Center
10662 Los Vaqueros Circle
P.O. Box 3014
Los Alamitos, CA 90720-1314
Tel: + 1 800 272 6657
Fax: + 1 714 821 4641
http://computer.org/cspress
csbooks@computer.org

IEEE Service Center
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
Tel: + 1 732 981 0060
Fax: + 1 732 981 9667
http://shop.ieee.org/store/
customer-service@ieee.org

IEEE Computer Society
Asia/Pacific Office
Watanabe Bldg., 1-4-2
Minami-Aoyama
Minato-ku, Tokyo 107-0062
JAPAN
Tel: + 81 3 3408 3118
Fax: + 81 3 3408 3553
tokyo.ofc@computer.org

Individual paper REPRINTS may be ordered at: <reprints@computer.org>

Editorial production by Bob Werner
Cover art production by Joe Daigle/Studio Productions
Printed in the United States of America by The Printing House

IEEE computer society

CPS Conference Publishing Services

IEEE Computer Society
***Conference Publishing Services* (CPS)**
http://www.computer.org/cps

Table of Contents

The Third International Conference on Availability, Reliability and Security (ARES 2008)

Session 3: Models

Session 4: Database

Session 5: Mobile

Session 6: RBAC and Recommender

Session 7: Risk Management

Session 8: Networks

Session 9: Software

Session 4: Networks

Session 5: Crypto

Session 6: Crypto and Health

Session 7: Models and Networks

Session 8: IDS

Session 9: Hardware

Workshop FARES

Session 1: Miscellaneous

Session 2: Access Control and Algorithms

Session 3: Crypto

Session 4: Risk Management

Workshop PSAI

Workshop APE

Workshop SECSE

Workshop DAWAM

Workshop WAIS

Workshop WSDF

Workshop SREIS

Chair's Message

The Third International Conference on Availability, Reliability and Security (ARES 2008 – The International Dependability Conference) brings together researchers and practitioners in the area of dependability. ARES 2008 highlights the various aspects of dependability, with special focus on the crucial linkage between availability, reliability and security.

ARES aims at a full and detailed discussion of research issues in the field of dependability as an integrative concept that covers amongst others availability, safety, confidentiality, integrity, maintainability and security and their different areas of applications.

This conference emphasizes the interplay between foundations and practical issues of dependability in areas such as information systems, e-government, m-government, location-based services, ubiquitous computing, and autonomous computing.

This years ARES conference is devoted to the critical examination and research challenges of the various aspects of Dependable Computing and the definition of a future road map.

We are very happy to welcome three well-known keynote speakers:

- Prof. Ravi Sandhu (Executive Director, Institute for Cyber-Security Research (ICSR) and Lutcher Brown Endowed Chair in Cyber-Security, University of Texas, San Antonio)
- Prof. Vijay Atluri (Management Science and Information Systems Department, Rutgers University)
- Prof. Günther Pernul (Department of Information Systems, University of Regensburg)

From over 200 submissions we have selected the 44 best for a presentation as full paper. In addition, several workshops and short papers show ongoing research projects and offer interesting starting points for discussions.

We wish all participants an enjoyable conference and interesting discussions.

Edgar R. Weippl, Secure Business Austria, Vienna University of Technology

Gerald Quirchmayr, University of Vienna and University of South Australia

Jill Slay, University of South Australia

Conference Officers

Honorary Co-Chairs
Roland Wagner, University of Linz, Austria

General Co-Chairs
Guenther Pernul, University of Regensburg, Germany
Makoto Takizawa, Tokyo Denki University, Japan

Program Co-Chairs
Gerald Quirchmayr, University of South Australia, Australia
Jill Slay, University of South Australia, Australia
Edgar Weippl, Vienna University of Technology / Secure Business Austria, Austria

Workshops Co-Chairs
Leonard Barolli, Fukuoka Institute of Technology (FIT), Japan
A Min Tjoa, Vienna University of Technology, Austria

Organizing Chair
Fatos Xhafa, Technical University of Catalonia, Spain

International Liaison Co-Chairs
Maria Wimmer, University of Koblenz-Landau, Germany
Charles Shoniregun, University of East London, United Kingdom

Publicity Chair
Vladimir Marik, Czech Technical University, Czech Republic

Implementing Multidimensional Security into OLAP Tools

Carlos Blanco[1], Eduardo Fernández-Medina[1], Juan Trujillo[2] and Mario Piattini[1]
[1]*Dep. of Information Technologies and Systems. Escuela Superior de Informática*
University of Castilla-La Mancha. Ciudad Real, Spain
{Carlos.Blanco, Eduardo.Fdzmedina, Mario.Piattini}@uclm.es
[2]*Department of Information Languages and Systems. Facultad de Informática*
University of Alicante. Alicante, Spain
jtrujillo@dlsi.ua.es

Abstract

Data Warehouses (DW) manage historical information for the decision making process and for enterprises, it is vitally important to consider security requirements from the earliest stages of the development process. OLAP tools are the most used tools for implementing and consulting DWs and it is necessary to define security measures to avoid that users can access unauthorized information by executing queries. We have created a MDA architecture for developing secure DW and in this paper, we will propose how to implement the security measures defined at upper abstraction levels using our approach into SQL Server Analysis Services 2008.

1. Introduction

Information security is a serious requirement that must be carefully taken into account, not as an isolated aspect, but as an element present in all stages of the development lifecycle, from requirement analysis to implementation and maintenance [1-3]. In this way, information assurance, security and privacy have moved from being considered by information systems designers as narrow topics of interest to become critical issues of fundamental importance in our society [4]. Some authors indicate that the survival of organizations depends on the correct management of information security and confidentiality [5].

On the other hand, multidimensional modeling is the foundation of Data Warehouses (DW), MD Databases and On-Line Analytical Processing Applications (OLAP). Data Warehouses (DW) use enterprise information for the decision making process and a user can find out very important information by using queries in OLAP tools. In this way, it is necessary that security measures defined in all early stages of the development process are applied in OLAP.

In addition, Model Driven Engineering (MDE) and the OMG Model Driven Architecture (MDA) [6] are model-oriented approaches for software development that are based on the separation between the specification of the system functionality and its implementation using specific platforms.

MDA defines a platform-independent model (PIM) that does not include information about specific platforms and technologies. This model (PIM) can be translated into one or more platform-specific models (PSM) with information about the used specific technology. Then, each PSM can be translated into a code that can be executed in the specific platform. There are several proposals for defining these translations between models [7], and OMG proposes to use Query/Views/Transformations (QVT) for defining transformations between models created by using Meta-Object Facility (MOF).

We have developed a proposal [8] for modeling secure Data Warehouses using a MDA approach that will be described in the following sections. This proposal does not deal with the implementation into OLAP tools. Therefore, in this paper we will analyze this problem and propose how security measures defined by using this approach can be finally implemented into OLAP tools. In future works, we will obtain this implementation in an automatic way following the MDA approach.

The rest of the paper is organized as follows: in Section 2 we will describe the background of this work; in Section 3 we will propose how to implement security requirements into OLAP tools. Finally, in Section 4 we will present our conclusions and future work.

0-7695-3102-4/08 $25.00 © 2008 IEEE
DOI 10.1109/ARES.2008.179

2. Background

In this section, first of all, we will describe the Model Driven Architecture for developing secure Data Warehouses; then, the Access Control and Audit model and finally what OLAP tools offer for defining security measures.

2.1. Model Driven Architecture for developing secure Data Warehouses

We have developed a *MDA architecture for developing secure Data Warehouses* [8]. Security rules specified at the conceptual level in the multidimensional model are translated into logical level and code. Figure 1 illustrates our architecture: the upper section presents the SMD CIM (Secure Multidimensional CIM) which defines both functional and non functional requirements for DWs at the business level. Through T1 transformation, we will obtain the SMD PIM (Secure Multidimensional PIM) at the conceptual level and through T2 transformation, we will obtain the SMD PSM (Secure Multidimensional PSM) at the logical level. Finally, through T3 we will obtain specific SMD code.

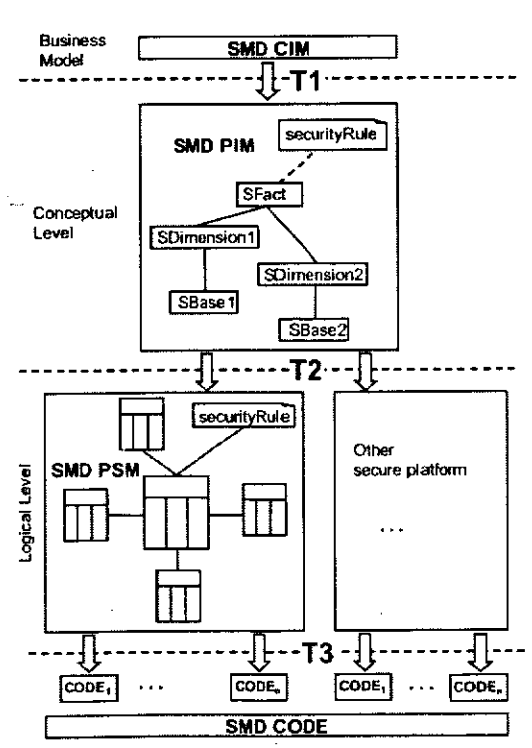

Figure 1. Model Driven Architecture for secure DW

The SMD PIM is defined by using an extension of the UML profile called Secure Data Warehouse (SECDW) [9] and it allows us to represent the main security requirements for DW conceptual modeling (an example is shown in Figure 2). This metamodel includes the main characteristics of Data Warehouses and security aspects defined according to our Access Control and Audit model. We can define security levels, user categories, user roles and security constraints for each element of the metamodel and we can define security rules (SIAR), authorization rules (AUR) and audit rules (AU) by using OCL expressions and UML notes associated with the corresponding class.

The specification of a platform-specific model is designed according to the specific properties of the Database Management Systems (DBMS). Our metamodel at the logical level for Secure Multidimensional PSM (SMD PSM) is a ROLAP approach called Secure Relational Data Warehouse (SECRDW) [10]. It extends the Relational Warehouse Metamodel (CWM) with security and audit capabilities.

In order to complete the MDA architecture, it is necessary to define the transformation between models. The transformation between conceptual and logical levels using SMD PIM (SECDW) and SMD PSM (SECRDW) has already been defined [11]. Furthermore, the transformation from SMD PSM to SMD code in a DBMS using Oracle Label Security has been defined too. Due to the fact that in Data Warehouses OLAP tools are more used than DBMS, our research effort is focused on developing secure code in OLAP tools according to the above-defined security requirements at conceptual and logical levels.

2.2. Access Control and Audit Model (ACA)

The ACA model [12] is an access control and audit model defined for DWs. Traditional access control models are based on relational concepts (tables, columns, rows, etc) and they are not appropriate for the multidimensional modeling used in data warehouses.

The ACA model is based on the conceptual model using our UML profile for DW [13] and allows us to specify security constraints in DW's multidimensional models. This model considers a combination of mandatory and role based access control which is based on the classification of subjects and objects in the system.

We can use levels, roles, compartments and security constraints to classify users. So, an *authorization subject* is an identity composed of: userID, user roles,

user compartments, security level or a levels interval and subjectExpression in OCL.

According to MD models, we can identify the main *authorization objects* as follows: facts, dimension, classification, hierarchy levels, measures, dimension attributes and instances. The authorization object component is composed of an identity (class name or attribute name) and an objectExpression in OCL.

Sensitive Information Assignment Rules (SIAR) specify multilevel security policies and allow us to define sensitivity information for each element in the MD model. The information we need for defining these rules is: objects, sensitivity information (security levels, user roles and user compartments), involved classes (some information can be confidential associated with other data) and a condition specified with OCL.

Authorization Rules (AUR) specify the subject which the rule applies to, the object which the authorization refers to, the action which the rule refers to (in this approach, we will only consider read operations) and the sign describing whether the rule permits or denies access. In Table 2, the syntax for authorization rules is shown.

Auditing rules (AR) help us ensure that authorized users do not misuse their privileges defining objects, log types (access to be recorded), log conditions and information to be logged.

2.3. Implementing security requirements into OLAP tools

Data Warehouses (DW) are usually implemented into OLAP tools. The main OLAP tools offer possibilities to define security measures but we can not define them at upper abstract levels in the multidimensional model. Our MDA architecture for developing secure DW includes security measures at CIM, PIM and PSM levels. In order to complete our architecture, we need to obtain secure code in OLAP tools that it is defined at business, conceptual and logic levels according to the security measures.

These OLAP tools work with operations that are usually the following ones: roll-up (increasing the level of aggregation) and drill-down (decreasing the level of aggregation) along one or more classification hierarchies, slice-dice (selection and projection) and pivoting (re-orienting the MD view of data which also allows us to exchange dimensions for facts; i.e., symmetric treatment of facts and dimensions).

For developing secure code, it is necessary to take into account how these OLAP operations work with information. Users can use these operations to obtain unauthorized information by using navigations or aggregations with operations or by inferring information with chained queries.

Current OLAP tools do not allow us to represent in a direct way all security measures that we can define by using our ACA model. On the one hand, Oracle Label Security 11g provides us with the possibility of defining security constraints by using levels, compartments and roles. We can grant and revoke permissions but not over multidimensional elements at upper abstraction levels as cubes or dimensions.

On the other hand, SQL Server Analysis Services (SSAS) 2008 offers us the possibility of working with multidimensional models as well as that of defining security measures over multidimensional elements. SSAS uses a role-based access control policy (RBAC) that is supposed to translate our measures defined according to our ACA model into role approach.

3. Implementing security requirements into Microsoft Analysis Services 2008

In this section, our security model will be implemented into an OLAP tool. SQL Server Analysis Services 2008 (SSAS) has been selected as the OLAP tool to be used for our implementation because it allows defining security measures in multidimensional structures.

3.1. Description of SASS 2008 security

Analysis Services security uses a role-based policy and each role contains one or more specific user accounts or user groups that we can use to establish restriction access measures for metadata and multidimensional elements such as cubes, dimension or stored values.

It lets us define security measures to establish administrative authorizations for processing a database, cube, dimension or mining structure and to establish read authorizations on metadata for showing or hiding definitions. We can to grant or revoke users access to data sources, cubes, cell data, dimensions, dimension data, and mining structures and define access control at the cell level by establishing allowed and denied sets using MDX queries.

Another interesting characteristic are the "Visual Totals" property that filters the aggregate values to calculate those using only the visible elements for the role considering the defined security measures; contingence permissions for read accesses to control inferences on derived cells and auditing capabilities that include audit objects, auditing DDL commands and support for multiple logging targets.

3.2. Methodology

As previously stated, SSAS offers us the possibility of defining security measures over multidimensional elements but it considers a role-based control access policy and establishes for each role which elements we can see. Next, we will analyze how to represent the security constraints defined using our ACA model in SSAS. According to our ACA model, subjects can be classified into levels, compartments and roles, and security constraints can be defined. As SSAS's policy is role-based we have to translate this security information by creating new roles for each possible classification as shown in Table 4.

ACA Model	SSAS 2008
Security Roles	For each security role "r" we create a "SRr" role.
Security Compartments	For each security compartment "c" we create a "SCc" role.
Security Levels	For each security level "l" we create a "SLl" role that contains users with this security level.
Security Constraints	Security constraints are specified when possible by defining denied sets using MDX queries. If it is necessary we will create a specific role "SCTx" that contains the unauthorized users.

Table 1. Role definition

In addition, we have to consider Sensitivity Information Assignment Rules (SIAR) and Authorization Rules (AUR) that have been defined in our model. We use an open policy with specific denials and these rules define multidimensional elements that are hidden for certain roles.

For each sensitivity information assignment rule (SIAR) we have to detect the involved roles as well as configure them by hiding certain DW elements that are denied for the considered rule. Each rule represents explicit denials for each role with certain security constraints and as we consider an open world policy, we will implement these constraints into SSAS by establishing MDX queries which hide these involved elements for certain roles.

We follow the same procedure for each authorization rules (AUR), detecting involved roles along with the multidimensional elements that have to be hidden for these roles by defining MDX queries.

3.3. Example

In this subsection, we will describe a complete example from its definition at upper abstract levels using our model to its implementation into SSAS.

Figure 2 shows our example at the conceptual level, modelled with our SECDW approach including sensitivity information assignment rules (SIAR) and authorization rules (AUR). In this example, we will define an enumeration with security levels: unclassified, confidential, secret and top secret. User roles follow the simplified hierarchy shown in Figure 3 and compartments have not been defined as they depend on organization policies.

3.3.1. Defining security roles. Firstly, we have to define roles for each possible security classification of the subjects. In this way, we will create in SSAS a specific role for each role of the hierarchy shown in Figure 3 (SRDoctor, SRNurse, SRHealth, etc) as well as for each security level (SLUnclassified, SLConfidential, SLSecret and SLTopsecret).

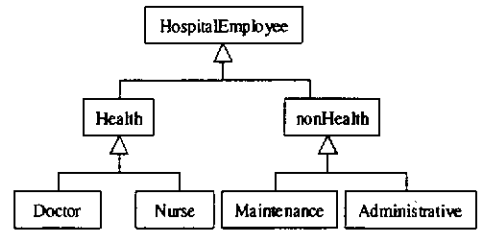

Figure 2. Roles hierarchy

3.3.2. Defining SIAR rules. Once the necessary roles have been defined, we have to translate the security rules.

The sensitivity information assignment rules (SIAR) defined as tagged values of classes or attributes are easy to translate because they do not present conditions and can be implemented by detecting involved roles and hiding affected DW elements.

SIAR titled "1" (Figure 2) specifies that "admission" class and involved classes have a top secret security level if the disease is "cancer" or "AIDS". Then, if the condition is satisfied we have the following explicit denial for each security level lower than top secret and for each involved class:

```
(-, SLSecret, CL Admission)
```

In SSAS, for each role that represents a security level lower than top secret (SLSecret, SLConfidential and SLUnclassified) we will hide all involved classes using the following MDX expression as denied set for each one:

```
Admission.Diagnosis.[Diagnosis_Group].&[cancer] or
Admission.Diagnosis.[Diagnosis_Group].&[AIDS]
```

SIAR titled "2" (Figure 2) defines that the "admission" class together with the "patient" involved

class have a top secret security level if the cost is higher than 10000. Then, if the condition is satisfied we will have the following denial for each involved role (with SL lower than secret) and class ("admission" and "patient"):

(-, SLSecret, CL Patient)

In SSAS, for each role that represents a security level lower than secret we will hide the involved classes using the following expression:

Admission.cost > 10000

3.3.3. Defining AUR rules. Now, we will translate the authorization rules (AUR).

AUR titled "4" (Figure 2) defines that queries involving "Diagnosis", "Diagnosis_Group" and "Patient" corresponding to a particular health area will be accessible to members of the user role Health only if their working area is the same as that health area. Note that this authorization rule is propagated up. It is necessary to define security constraints on SRHealth role including the following MDX expression as denied set for each involved classes:

Diagnosis.healthArea<>currentUser.userProfile.WorkingArea

AUR titled "5" (Figure 2) defines that Patients will be special users of the system as we would like them to have access to their own data. Each patient will therefore have permission to access his/her own data (not to other patients' data) when querying classes Patient, Admission, and Diagnosis.

As this authorization rule is positive, it is propagated down. In this case it is better to define just one positive rule for all users using "SRHospitalEmployee" role. We will establish the following expression as allowed set in this role:

currentUser.UserProfile.name = Patient.name

3.3.4. Defining AR rules. SSAS allows us to audit activity on our data including information about when data has been read and modified. To establish these rules we can define audit specification in data warehouses with SQL Server 2008.

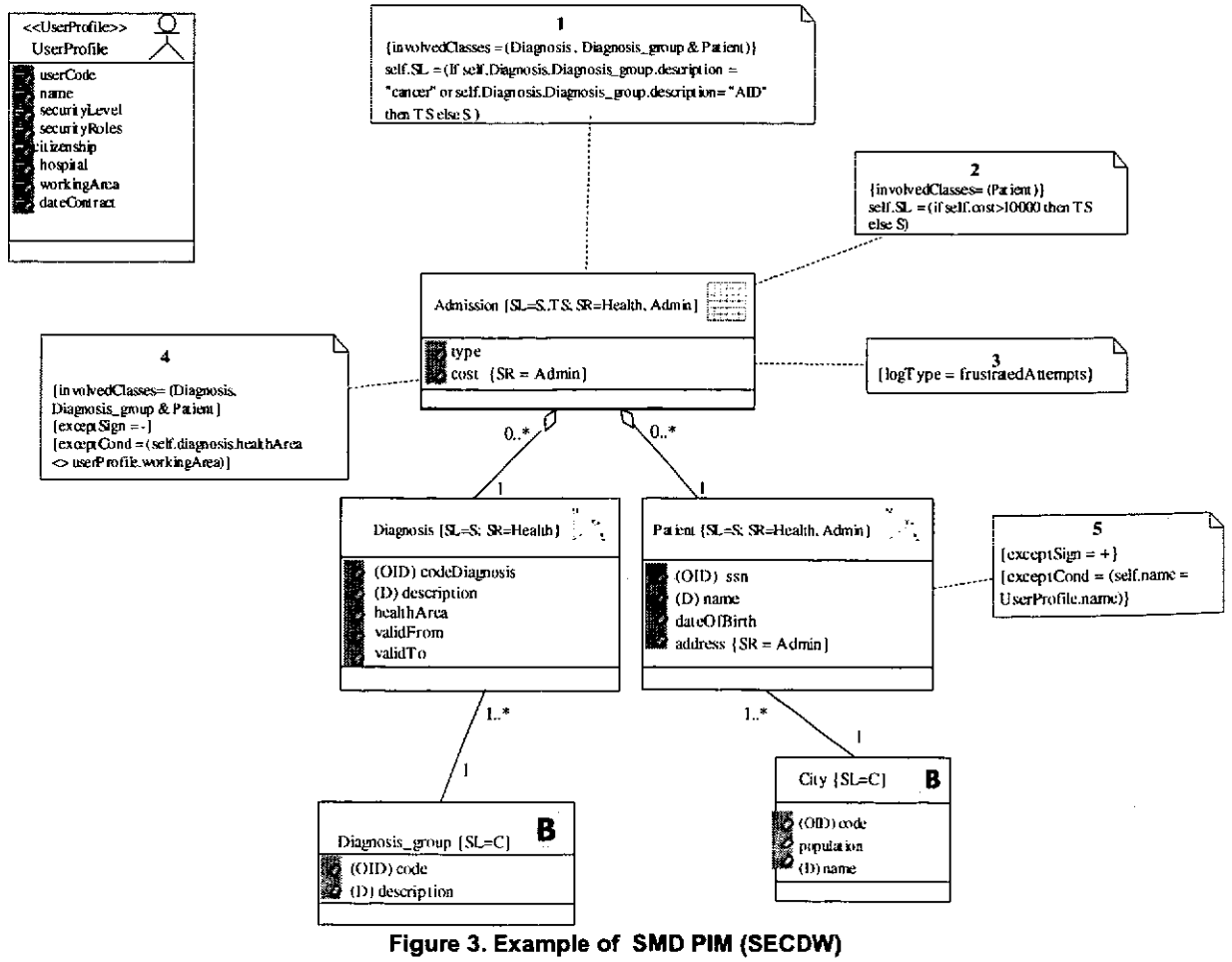

Figure 3. Example of SMD PIM (SECDW)

4. Conclusions

The first conclusion of our work is that OLAP tools do not support the complete security requirements definition over the multidimensional model at upper abstraction levels. These tools deal with partial security establishment, i.e. SSAS uses RBAC as access control policy, but they do not consider how OLAP operations affect to the defined security constraints. Users could access unauthorized information by using operations that navigate over the multidimensional structures.

These tools allow us to set security constraints over data or users but not over OLAP operations at upper abstraction levels. It is necessary that navigability constraints can be represented at the conceptual levels and finally respected in OLAP tools. Another problem is the hiding of the existence of determinate hide data. SSAS gives us null or N/A values but we know that this information exists and we have tips to obtain it.

In addition, the inference problem is not completely supported and requires a lot of computational effort to control. There are some strategies to control inferences; for instance, to limit the maximum number of sub queries or to include perturbations in the data. SSAS allows us to control inferences over derivative data but not other that involving queries. We can define contingence chains that represent dependences between data to denied data access if you do not have permissions over all data of which it is derived from.

When we have translated the security requirements defined at the conceptual level into OLAP code, we have found that our security model is richer than the security capabilities that OLAP tools offer. To define security in SSAS we have to adapt our security information in a role-based approach and to represent rules we have to hide multidimensional elements for each involved role. This is an iterative and tedious process and it would be interesting that OLAP tools allow us to define security constraints over multidimensional elements. However, the biggest detected problem is the impossibility of representing security constraints on OLAP operations at upper abstraction levels. These constraints would allow us to avoid the access to unauthorized information by using navigations or inferences.

Our MDA architecture for developing secure DW allows us to define security requirements but should be extended with the possibility of establishing navigations and inferences constraints that can be translated into OLAP code. In the same way, tools should be extended to give us control over navigations and inferences that can find out unauthorized enterprise information.

Acknowledgment

This research is part of the Projects ESFINGE (TIN2006-15175-C05-05) financed by the Spanish Ministry of Education and Science, and the MISTICO (PBC-06-0082), DADS (PBC-05-012-2) and DIMENSIONS (PBC-05-012-2) financed by the FEDER and the Regional Science and Technology Ministry of Castilla-La Mancha (Spain).

References

[1] Devanbu, P. and S. Stubblebine, *Software engineering for security: a roadmap*. ACM Press. Future of Software Engineering, 2000: p. 227-239.

[2] Ferrari, E. and B. Thuraisingham. *Secure Databases Systems*. in *Advanced Databases: Technology Design*. 2000. Artech Huse: London.

[3] Mouratidis, H. and P. Giorgini, *An Introduction*, in *Integrating Security and Software Engineering: Advances and Future Visions*. 2006, Idea Group Publishing.

[4] Denker, G., L. Kagal, and T. Finin, *Security in the Semantic Web using OWL*. Information Security Technical Report, 2005. 10(1): p. 51-58.

[5] Dhillon, G. and J. Backhouse, *Information system security management in the new millennium*. Communications of the ACM, 2000. 43(7): p. 125-128.

[6] MDA, O.M.G., *Model Driven Architecture Guide*. 2003.

[7] Czarnecki, K. and S. Helsen, *Classification of model transformation approaches.*, in *2nd OOPSLA Workshop on Generative Techniques in the Context of the Model Driven Architecture*. 2003.

[8] Fernández-Medina, E., J. Trujillo, and M. Piattini, *Model Driven Multidimensional Modeling of Secure Data Warehouses*. European Journal of Information Systems, 2007. 16: p. 374-389.

[9] Fernandez-Medina, E., et al., *Developing secure data warehouses with a UML extension*. Information Systems, 2007. 32(6): p. 826-856.

[10] Soler, E., et al. *Representing Security and Audit Rules for Data Warehouses at the Logical Level by using the Common Warehouse Metamodel*. in *1st Int. Conference on Availability, Reliability and Security*. 2006. Vienna, Austria.

[11] Soler, E., et al. *A Set of QVT relations to Transform PIM to PSM in the Design of Secure Data Warehouses*. in *IEEE International Symposium on Frontiers on Availability, Reliability and Security (FARES 2007)*. 2007. Viena, Austria.

[12] Fernandez-Medina, E., et al., *Access control and audit model for the multidimensional modeling of data warehouses*. Decision Support Systems, 2006. 42(3): p. 1270-1289.

[13] Lujan-Mora, S., J. Trujillo, and I.-Y. Song, *A UML profile for multidimensional modeling in data warehouses*. Data & Knowledge Engineering, 2006. 59(3): p. 725-769.

Author Index

The Third International Conference on Availability, Reliability and Security (ARES 2008)

Notes

IEEE Computer Society Conference Publications Operations Committee

IEEE Computer Society Publications

The world-renowned IEEE Computer Society publishes, promotes, and distributes a wide variety of authoritative computer science and engineering texts. These books are available from most retail outlets. Visit the CS Store at *http://www.computer.org/portal/site/store/index.jsp* for a list of products.

IEEE Computer Society *Conference Publishing Services* (CPS)

The IEEE Computer Society produces conference publications for more than 250 acclaimed international conferences each year in a variety of formats, including books, CD-ROMs, USB Drives, and on-line publications. For information about the IEEE Computer Society's *Conference Publishing Services* (CPS), please e-mail: cps@computer.org or telephone +1-714-821-8380. Fax +1-714-761-1784. Additional information about *Conference Publishing Services* (CPS) can be accessed from our web site at: *http://www.computer.org/cps*

IEEE Computer Society / Wiley Partnership

The IEEE Computer Society and Wiley partnership allows the CS Press *Authored Book* program to produce a number of exciting new titles in areas of computer science and engineering with a special focus on software engineering. IEEE Computer Society members continue to receive a 15% discount on these titles when purchased through Wiley or at: *http://wiley.com/ieeecs*. To submit questions about the program or send proposals, please e-mail jwilson@computer.org or telephone +1-714-816-2112. Additional information regarding the Computer Society's authored book program can also be accessed from our web site at:
http://www.computer.org/portal/pages/ieeecs/publications/books/about.html

Revised: 21 January 2008

CPS Online™

IEEE Online Collaborative Publishing Environment

CPS Online is our innovative online collaborative conference publishing system designed to speed the delivery of price quotations and provide conferences with real-time access to all of a project's publication materials during production, including the final papers. The ***CPS Online*** workspace gives a conference the opportunity to upload files through any Web browser, check status and scheduling on their project, make changes to the Table of Contents and Front Matter, approve editorial changes and proofs, and communicate with their CPS editor through discussion forums, chat tools, commenting tools and e-mail.

The following is the URL link to the ***CPS Online*** Publishing Inquiry Form:
http://www.ieeeconfpublishing.org/cpir/inquiry/cps_inquiry.html

Published by the IEEE Computer Society
10662 Los Vaqueros Circle
P.O. Box 3014
Los Alamitos, CA 90720-1314

IEEE Computer Society Order Number P3102
Library of Congress Number 2007909935
ISBN 0-7695-3102-4