cture Notes in Computer Science

e LNCS series reports state-of-the-art results in computer science
search, development, and education, at a high level and in both printed
d electronic form. Enjoying tight cooperation with the R&D community,
th numerous individuals, as well as with prestigious organizations and
cieties, LNCS has grown into the most comprehensive computer science
search forum available.

e scope of LNCS, including its subseries LNAI and LNBI, spans the
ole range of computer science and information technology including
erdisciplinary topics in a variety of application fields. The type of
terial published traditionally includes

roceedings (published in time for the respective conference)
ost-proceedings (consisting of thoroughly revised final full papers)
esearch monographs (which may be based on outstanding PhD work,
esearch projects, technical reports, etc.)

re recently, several color-cover sublines have been added featuring,
yond a collection of papers, various added-value components; these
olines include

utorials (textbook-like monographs or collections of lectures given at
dvanced courses)
tate-of-the-art surveys (offering complete and mediated coverage
f a topic)
ot topics (introducing emergent topics to the broader community)

parallel to the printed book, each new volume is published electronically
LNCS Online.

tailed information on LNCS can be found at
w.springer.com/lncs

posals for publication should be sent to
CS Editorial, Tiergartenstr. 17, 69121 Heidelberg, Germany
nail: lncs@springer.com

N 0302-9743
SBN 978-3-642-15545-1
783642 155451

Lecture Notes in Computer Science
LNCS LNAI LNBI

Jonker • Petković (Eds.)

LNCS 6358

LNCS 6358

Secure Data Management

Willem Jonker
Milan Petković (Eds.)

Secure Data Management

7th VLDB Workshop, SDM 2010
Singapore, September 2010
Proceedings

Springer

Lecture Notes in Computer Science 6358

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Willem Jonker Milan Petković (Eds.)

Secure Data Management

7th VLDB Workshop, SDM 2010
Singapore, September 17, 2010
Proceedings

Springer

Volume Editors

Willem Jonker
Philips Research Europe
High Tech Campus 34, 5656 AE Eindhoven, The Netherlands
E-mail: willem.jonker@philips.com
and
University of Twente
Department of Computer Science
Enschede, The Netherlands
E-mail: jonker@cs.utwente.nl

Milan Petković
Philips Research Europe
High Tech Campus 34, 5656 AE Eindhoven, The Netherlands
E-mail: Milan.Petkovic@philips.com
and
Eindhoven University of Technology
Faculty of Mathematics and Computer Science (HG 9.85)
Eindhoven, The Netherlands
E-mail: m.petkovic@tue.nl

Library of Congress Control Number: 2010933962

CR Subject Classification (1998): K.6.5, D.4.6, E.3, C.2, H.4, H.3

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN 0302-9743
ISBN-10 3-642-15545-6 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-15545-1 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper 06/3180

Preface

The VLDB Secure Data Management Workshop was held for the 7th time this year.

The topic of data security remains an important area of research especially due to the growing proliferation of data in open environments as a result of emerging data services such as cloud computing, location based services, and health-related services. Confidentiality is the main driving force behind the research that covers topics such as privacy enhancing technologies, access control, and search in encrypted data.

We received 20 submissions from which the program committee selected 10 papers to be presented at the workshop and included in the proceedings (50% acceptance rate). In addition, we are proud that Elisa Bertino accepted our invitation to give a keynote for which she selected the topic of data trustworthiness. We hope the papers collected in this volume will stimulate your research in this area.

The regular papers in the proceeding have been grouped into two sections. The first section focuses on privacy. The papers in this section present a balanced mix of theoretical work on anonymity and application-oriented work.

The second section focuses on data security in open environments. The papers address issues related to the management of confidential data that is stored in or released to open environments, such as, for example, in cloud computing.

We wish to thank all the authors of submitted papers for their high-quality submissions. We would also like to thank the program committee members as well as additional referees for doing an excellent review job. Finally, let us acknowledge the work of Luan Ibraimi, who helped in the technical preparation of the proceedings.

July 2010

Willem Jonker
Milan Petković

Organization

Workshop Organizers

Willem Jonker	Philips Research/University of Twente, The Netherlands
Milan Petković	Philips Research/Eindhoven University of Technology, The Netherlands

Program Committee

Gerrit Bleumer	Francotyp-Postalia, Germany
Ljiljana Brankovic	University of Newcastle, Australia
Sabrina De Capitani di Vimercati	University of Milan, Italy
Ernesto Damiani	University of Milan, Italy
Eric Diehl	Technicolor, France
Lee Dong Hoon	Korea University, South Korea
Jeroen Doumen	Irdeto, The Netherlands
Csilla Farkas	University of South Carolina, USA
Elena Ferrari	University of Insubria, Italy
Simone Fischer-Hübner	Karlstad University, Sweden
Tyrone Grandison	IBM Almaden Research Center, USA
Dieter Gollmann	Technische Universität Hamburg-Harburg, Germany
Marit Hansen	Independent Centre for Privacy Protection, Germany
Min-Shiang Hwang	National Chung Hsing University, Taiwan
Mizuho Iwaihara	Kyoto University, Japan
Sushil Jajodia	George Mason University, USA
Ton Kalker	HP Labs, USA
Marc Langheinrich	Università della Svizzera italiana (USI), Switzerland
Nguyen Manh Tho	Vienna University of Technology, Austria
Nick Mankovich	Philips Medical Systems, USA
Sharad Mehrotra	University of California at Irvine, USA
Stig Frode Mjølsnes	Norwegian University of Science and Technology, Norway
Eiji Okamoto	University of Tsukuba, Japan
Sylvia Osborn	University of Western Ontario, Canada
Günther Pernul	University of Regensburg, Germany
Birgit Pfitzmann	IBM Watson Research Lab, Switzerland

Bart Preneel	K.U.Leuven, Belgium
Kai Rannenberg	Goethe University Frankfurt, Germany
Andreas Schaad	SAP Labs, France
Nicholas Sheppard	Queensland University of Technology, Australia
Jason Smith	Queensland University of Technology, Australia
Morton Swimmer	John Jay College of Criminal Justice/CUNY, USA
Clark Thomborson	University of Auckland, New Zealand
Sheng Zhong	State University of New York at Buffalo, USA

Additional Referees

Mamadou Diallo	University of California at Irvine, USA
Christian Kahl	Goethe University Frankfurt, Germany
Christian Weber	Goethe University Frankfurt, Germany

Table of Contents

2. Baek, J., Safavi-Naini, R., Susilo, W.: Public key encryption with keyword search revisited. Cryptology ePrint Archive, Report 2005/191 (2005), http://eprint.iacr.org/
3. Bellare, M., Boldyreva, A., O'Neill, A.: Deterministic and efficiently searchable encryption. In: Menezes [16], pp. 535–552
4. Boneh, D., Di Crescenzo, G., Ostrovsky, R., Persiano, G.: Public key encryption with keyword search. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 506–522. Springer, Heidelberg (2004)
5. Boneh, D., Kushilevitz, E., Ostrovsky, R., Skeith III, W.E.: Public key encryption that allows PIR queries. In: Menezes [16], pp. 50–67
6. Chang, Y.-C., Mitzenmacher, M.: Privacy preserving keyword searches on remote encrypted data. In: Ioannidis, J., Keromytis, A.D., Yung, M. (eds.) ACNS 2005. LNCS, vol. 3531, pp. 442–455. Springer, Heidelberg (2005)
7. Chor, B., Gilboa, N.: Computationally private information retrieval (extended abstract). In: STOC, pp. 304–313 (1997)
8. Chor, B., Kushilevitz, E., Goldreich, O., Sudan, M.: Private information retrieval. J. ACM 45(6), 965–981 (1998)
9. Curtmola, R., Garay, J.A., Kamara, S., Ostrovsky, R.: Searchable symmetric encryption: improved definitions and efficient constructions. In: Juels, A., Wright, R.N., di Vimercati, S.D.C. (eds.) ACM Conference on Computer and Communications Security, pp. 79–88. ACM, New York (2006)
10. Goh, E.-J.: Secure indexes. Cryptology ePrint Archive, Report 2003/216 (2003), http://eprint.iacr.org/
11. Goldreich, O., Goldwasser, S., Halevi, S.: Public-key cryptosystems from lattice reduction problems. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 112–131. Springer, Heidelberg (1997)
12. Goldreich, O., Ostrovsky, R.: Software protection and simulation on oblivious rams. J. ACM 43(3), 431–473 (1996)
13. Goldwasser, S., Micali, S.: Probabilistic encryption. J. Comput. Syst. Sci. 28(2), 270–299 (1984)
14. Kushilevitz, E., Ostrovsky, R.: Replication is not needed: Single database, computationally-private information retrieval. In: FOCS, pp. 364–373 (1997)
15. Lamport, L.: Password authentification with insecure communication. Commun. ACM 24(11), 770–772 (1981)
16. Menezes, A. (ed.): CRYPTO 2007. LNCS, vol. 4622. Springer, Heidelberg (2007)
17. Song, D.X., Wagner, D., Perrig, A.: Practical techniques for searches on encrypted data. In: IEEE Symposium on Security and Privacy, pp. 44–55 (2000)
18. Troncoso-Pastoriza, J.R., Katzenbeisser, S., Celik, M.U.: Privacy preserving error resilient dna searching through oblivious automata. In: Ning, P., di Vimercati, S.D.C., Syverson, P.F. (eds.) ACM Conference on Computer and Communications Security, pp. 519–528. ACM, New York (2007)

Towards the Secure Modelling of OLAP Users' Behaviour

Carlos Blanco[1], Eduardo Fernández-Medina[2], Juan Trujillo[3], and Jan Jurjens[4]

[1] Dep. Of Mathematics, Statistical and Computation. Facultad de Ciencias
University of Cantabria. Av. De los Castros s/n. 39071. Santander. Spain
Carlos.Blanco@unican.es
[2] Dep. of Information Technologies and Systems. Escuela Superior de Informática.
GSyA Research Group. University of Castilla-La Mancha.
Paseo de la Universidad, 4. 13071. Ciudad Real. Spain
Eduardo.Fdezmedina@uclm.es
[3] Dep. of Information Languages and Systems. Facultad de Informática.
LUCENTIA Research Group. University of Alicante. San Vicente s/n. 03690.
Alicante. Spain
jtrujillo@dlsi.ua.es
[4] Germany TU Dortmund & Fraunhofer ISST. Germany
jan.jurjens@cs.tu-dortmund.de

Abstract. Information Security is a crucial aspect for organizations, and must be considered during the development of Information Systems. The data in Data Warehouses (DWs) are highly sensitive since they manage historical information which is used to make strategic decisions, and security constraints should therefore be included in DW modelling within its structural aspects. However, another dynamic security component is also related to the sequences of OLAP (On-Line Analytical Processing) operations, and could be used to access (or infer) unauthorized information. This paper complements the modelling of DWs with state models, which permit the modelling of these dynamic situations in which sensitive information could be inferred. That is, it models queries that include security issues, and controls that their evolution through the application of OLAP operations always leads to authorized states. Finally, our proposal has been applied to a healthcare case study in which a DW manages admissions information with various security constraints.

Keywords: Data Warehouses, OLAP, Users Behaviour, Query Evolution, State Models, Security, Inference, Healthcare.

1 Introduction

DWs organize enterprises' historical information, which originates in heterogeneous data sources, for the decision-making process. It is widely accepted that the information in DWs is organized on the basis of multidimensional modelling in which facts represent the interesting measures of a business process to be analyzed (e.g. "sales", "deliveries", etc.) and related dimensions classify this information by the subjects that

represent the context in which a fact is analysed (e.g. "product", "customer", "time", etc.) [1, 2].

Information security is a critical aspect which must be considered during the entire Information Systems development process [3-8]. Security requirements can therefore be identified from the early stages of the development process and taken into consideration in design decisions, thus ensuring their perfect fit in a better quality solution.

Since DWs manage highly sensitive information which supports the decision making process and which also usually includes personal data protected by legal regulations, security and privacy are absolutely vital and should be considered in all layers and operations of the DW from the beginning, as strong requirements to their final implementation in DBMS (Data Bases Management Systems) or OLAP (On-Line Analytical Processing) tools [9].

Several proposals for DW modelling through a consideration of their specific structural characteristics (facts, dimensions, bases, hierarchies, etc.) exist, but only some of them include security aspects in their modelling [10-12]. However, these contributions deal with the security problem in a static manner in which a set of security constraints basically establish what information will be shown to or hidden from the user, depending on his/her security profile.

Our previous work has been focused on the static modelling of DWs by defining several models that have been improved with security capabilities. This has been aligned with a model driven architecture, thus permitting modelling at the business (CIM), conceptual (PIM) and logical (PSM) levels, and the automatic generation of final implementation into DBMS or OLAP tools by applying transformation rules. Our complete proposal was applied in a healthcare case study in which the "admissions" to a hospital were analyzed by studying patients and diagnosis, and considering several security constraints defined over sensitive information [11, 13-17].

Nevertheless, DW confidentiality could also be compromised by sequences of OLAP operations (such us drill-down or roll-up) which could be carried out by a malicious user in order to access or to infer unauthorized information, and these security constraints cannot be defined in a static model because they depend on the evolution of the queries.

Since the modelling of all possible queries is not financially viable, our proposal is focused on sensitive queries (sensitive joints of information) and their evolution. In a first stage, sensitive queries are detected and included in the static model by using a new kind of security rule called a "Joint Rule" which specifies the security privileges needed to combine this information.

In a second stage, the sensitive queries are then modelled by using state models. Possible evolutions of the query are represented as states which are reached by applying certain OLAP operations. The designer establishes security constraints to decide which states can be reached and what information should be shown or hidden depending on the user's security privileges and the previously visited states, thus avoiding the discovery of unauthorized information.

This paper is organized as follows: Section 2 presents our proposal for the modelling of secure OLAP systems by focusing on the secure modelling of OLAP users' behaviour through the use of state diagrams; Section 3 provides a case study; and finally, Section 4 shows our conclusions and future work.

2 Secure Modelling of OLAP Systems

Figure 1 shows an overview of our proposal to model secure OLAP systems. In a first stage, DW static aspects are modelled by using a UML profile [18]. Sensitive queries and their evolution through the application of OLAP operations are then modelled by using state models which fulfil the static model. And finally, the user's session is also controlled in order to avoid inferences between different queries.

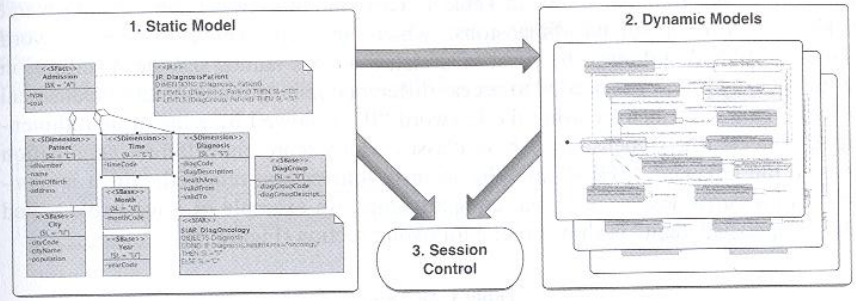

Fig. 1. Security Modelling Proposal for OLAP Systems

2.1 Static Modelling

A UML profile which is specifically created for DWs [18] allows the DW to be modelled at a conceptual abstraction level. On the one hand, structural aspects can be defined (such as fact, dimension and base classes, measures, attributes, hierarchies, etc.) and on the other hand, it is complemented with security constraints by using an Access Control and Audit (ACA) model [15]. This ACA model uses a classification consisting of three points of view: security levels (the users' clearance level), compartments (horizontal classification) and roles (hierarchical structure). It also permits security information (security levels, compartments and roles) to be associated for each element in the model (fact, dimension, etc.), and provides a set of security rules with which to define sensitive information, authorizations and auditing information.

Sensitive information assignment rules (SIAR) and authorization rules (AUR) are responsible for hiding certain information depending on the security profile (security level, compartment and role) of the user who attempts to access it.

- These rules are solved in design time, that is, the elements (facts, dimensions, bases, attributes, etc.) that have to be hidden in each user profile are already known. For instance, the "Diagnosis" dimension will be hidden from users with a security level that is lower than "Secret" (see Figure 3).
- In other situations, these rules are more complex and include conditions which have to be evaluated in execution time in order to hide the instances which do not satisfy them. For example, instances of "Diagnosis" with a health area attribute that is equal to "oncology" will require a security level of "Top Secret" (Figure 3).

Nevertheless, although the security privileges needed to access each element separately (e.g. each dimension) have been defined in the static model, the combination of several information elements is usually more sensitive than when they are accessed separately and an additional specification is required for these more restrictive security privileges (for example, in order to access "Patients" in relation to their "Diagnosis").

This paper includes a new kind of security rules, denominated as Joint Rules, to establish the security privileges needed to access certain combinations of information. Joint Rules (JR) are associated with a fact class in the model, and must be defined according to the grammar shown in Table 1. This kind of rules is composed of several fields: a set of involved dimensions, which are represented by the keyword "DIMENSIONS", followed by a list of dimension class names and the specification of the security privileges needed to access different combinations of multidimensional objects. This is defined by using the keyword "IF" followed by a list of multidimensional objects (the dimension or base classes which represent different aggregation levels), the keyword "THEN" and the security information required, which is composed of security levels, roles and compartments. An example of a joint rule called "JR_DiagnosisPatient" is shown in the following section (Figure 3).

Table 1. JR Syntax

JR := INVDIMENSIONS DEFLEVEL+
INVDIMENSIONS := "DIMENSIONS" SDimensionClassName+
DEFLEVEL := "IF (" OBJECT+ ") THEN" SECINF
OBJECT := SDimensionClassName | SBaseClassName
SECINF := SLEVEL? SROLES? SCOMPARTMENTS?
SLEVEL := "SL=" securityLevel
SROLES := "SR=" userRole+
SCOMPARTMENTS := "SC=" userCompartment+

2.2 Dynamic Modelling

In this section, dynamic models (state models) are proposed in order to enrich the aforementioned static models by including security aspects which are related to queries and their evolution through the application of OLAP operations and cannot, therefore, be modelled in a static manner.

Queries involve the combination of several dimensions at certain granularity levels, and this combination tends to be more sensitive than the separate accessing of data. These sensitive combinations are detected and modelled in the static model by using the new kind of security rules - joint rules - proposed in this paper, but their evolution through the application of OLAP operations should also be considered in order to ensure that confidentiality is not compromised. Thus, once sensitive queries have been specified by using joint rules, there is a dynamic modelling stage in which each sensitive query is modelled by using an extension of state models for secure OLAP systems.

Figure 2 shows an overview of our modelling proposal. States represent evolutions of a query obtained after applying certain OLAP operations that change the aggregation level (drill down and roll up).

Each state defines which pieces of information will be provided to the user. Firstly, when a state is reached, an entry action executes a slice operation with a set of restrictions which hide the unauthorized information. A dice operation then selects the multidimensional elements (measures, dimensions, bases, etc.) and members which should be shown according to the user's privileges.

The actions which can be achieved by users to allow them to navigate towards the hierarchies involved (drill down and roll up operations) establish connections between states and define the security guard conditions (security privileges) needed to reach them. For instance, the example shown in Figure 2 considers a hierarchy "A-B". If a user in "State 1" wants to show "B", that a user's security privileges will be checked ("securityGuard"), and if they are satisfied, a drill down operation will be achieved. "State 2" is then reached and the specified slice and dice operations are carried out.

Fig. 2. Secure state model overview

The starting point of the diagram leads to the less sensitive state that defines a query involving the dimensions specified in the joint rule grouped by the less restrictive aggregation level. Since the user can finish his/her session in any time, a specific end point has not been established.

In this approach, the combination of several multidimensional elements in a certain aggregation level signifies that it is necessary to create several states with different visibility privileges. These states are created by considering: (1) the security rules defined in the static model, including the new kind of security rules (joint rules); and (2) the designer's decisions which, according to the previously visited states, establish what information should be shown to guarantee confidentiality.

The designer defines the slices and dices for each state by using multidimensional expressions (MDX). For instance, a condition with which to hide oncology diagnosis included in a SIAR can be expressed in a slice operation as "diagnosis.healthArea<>"oncology"". There are some interesting expressions for the security point of view which allow certain dimension members to be shown: all members,

Table 2. MDX security expressions defined

Expression	Description
x.AllMembers	shows all the instances of a dimension or base "x"
x.VisibleMembers	after applying the specified visibility restrictions (slices), only shows the visible instances of a dimension or base "x"
x.NonEmptyMembers	after applying the specified visibility restrictions (slices), only shows the instances of a dimension or base "x" which contain data since, in some situations, showing empty members could be used to infer information
Null	hides all the instances

visible members, non-empty members or none of them. Some of them can be easily expressed in MDX, such as all members (".allMembers"), but others, such as non-empty members, require more complex expressions. Since the use of these expressions will be very usual in the secure dynamic modelling, we have defined a set of them (see Table 2).

2.3 Session Control

In the previous stages, the security rules solved in design time and in execution time (evaluating conditions) have been defined in the static models, and sensitive combinations of information have also been detected (by using joint rules) and modelled (by using state models) in order to control the evolution of queries by applying OLAP operations.

Since users could achieve a sequence of different queries, users' sessions should also be controlled in order to detect the possibility of an information inference. For example, although the combination of "Patients" and "Diagnosis" was established as being sensitive, an attacker could query "Patients" with "Time" and then "Diagnosis" with "Time", and thus infer unauthorized information by crossing data. This problem could be modelled by using this proposal, i.e., by using additional joint rules and static models, but we believe that it is of greater interest to control inferences by analyzing users' sessions.

Each session is composed of several events which are different queries:

$$Session_n = <e_1, e_2, \ldots e_n>$$

This session control stage checks each event and uses a stack to store the multidimensional elements which have been shown. It then uses the joint rules defined in the static model to discover what the sensitive combinations (sensitive joints) are, and analyses the stack to find the possibility of an inference.

The administrator is informed if the possibility of an inference has been detected and he/she decides what is the best action to take: to introduce noise in the data; to deny certain queries; to reduce the privileges of certain users; etc.

3 Case Study

In previous works, our secure data warehouses modelling proposal was applied in a healthcare case study by using static models in which both the structural and security aspects of data warehouses could be defined [11, 13-17]. The secure state models presented in this paper complement our proposal by dealing with the dynamic security problems that could not be modelled with a static approach. This case study shows how our dynamic proposal is applied to a healthcare system for a DW that analyzes hospital admissions, and manages information concerning diagnosis, patients and time.

Firstly, Figure 3 shows the conceptual model for our case study, defined by using the UML profile presented in [18]. This example includes a secure fact class "Admission" with two measures: "type" and "cost". Admissions are classified by using three

dimension classes: "Diagnosis", "Patient" and "Time", which are related to several base classes, thus creating three classification hierarchies with different aggregation levels ("Diagnosis-DiagnosisGroup", "Patient-City" and "Time-Month-Year").

The security configuration used is a sequence of security levels (Top Secret "TS", Secret "S", Confidential "C" and Undefined "U") and a hierarchy of security roles (a root role Employee "E" which has two sub-roles: Health "H" and Admin "A").

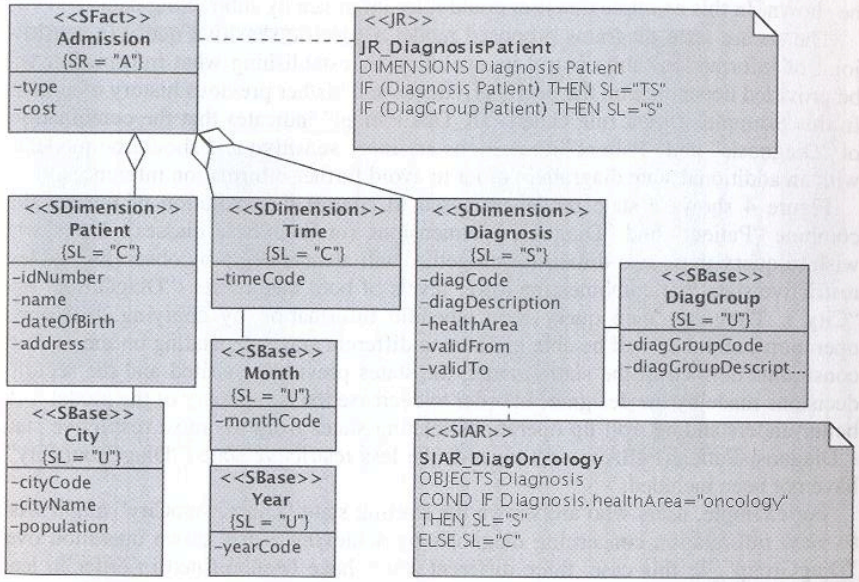

Fig. 3. Conceptual model

Various sensitive information assignment rules (SIAR) have also been defined. Some of these are directly represented by using stereotypes, such as the "Diagnosis" dimension, which uses the stereotype "SL=C" to establish the security level necessary to access the information.

On the other hand, a more complex SIAR called "SIAR_DiagOncology" assigns a higher security level ("Secret") to "Diagnosis" if the health area is oncology. This has been defined by using an OCL expression in a note associated with the "Diagnosis" dimension class.

An example of the new kind of security rule proposed in this paper, Joint Rule (JR), has been also defined. The rule "JR_DiagnosisPatient" indicates that the querying of the "Diagnosis" and "Patient" dimensions together is more sensitive and requires more restrictive security privileges. Since there are different aggregation levels in the hierarchies involved, the security privileges needed for each sensitive combination are specified separately in the rule. In this example, the combination of "Diagnosis" and "Patient" levels requires a "Top Secret" security level, whereas the combination of "DiagnosisGroup" and "Patient" levels requires "Secret".

Although security constraints have been considered in the design of the static model, if this model is not complemented with dynamic models, confidentiality could be compromised when users launch a query and apply a sequence of OLAP operations. For instance, a user could query "cost" information relating cities ("City" base class) with diagnosis groups ("DiagGroup" base class) and obtain a sum of costs. He/she could then apply a drill down operation over the diagnosis groups and obtain a different sum of costs because his/her security privileges do not allow all the data to be shown. In this example this user could infer inference by subtracting data.

The secure state diagrams proposed model a specific sensitive query (a sensitive joint of information) and control its evolution by establishing what information will be provided depending on the user's privileges and his/her previous history of actions. In this example, a joint rule called "JR_DiagPatient" indicates that the combinations of "Diagnosis" and "Patient" dimensions are more sensitive and should be modelled with an additional state diagram in order to avoid further information inferences.

Figure 4 shows a state model proposed to control the evolution of queries that combine "Patient" and "Diagnosis" dimensions (or their base classes). Users who wish to query these two dimensions together will always begin by observing the less restrictive state that combines the lower levels of both hierarchies ("DiagGroup" and "City"). They can then query more specific information by applying drill down operations, and they will be able to move to different states depending on the security constraints defined in the static model, the states previously visited and the security decisions made by the designer. In order to decrease the complexity of the model for a better understanding, roll up operations relating states from the most restrictive state ("DiagnosisPatient-FullAccess") towards the less restrictive state ("DiagGroupCity") have not been included.

For example, users who are shown the starting state "DiagGroupCity" might wish to view information concerning diagnosis by achieving a drill down operation over "DiagGroup". In this case, three different states have been defined in order to lead each kind of user towards a secure state in which confidentiality is not compromised:

- Users with a security level of "Top Secret" go to the "DiagnosisCity-FullAccess" state which provides them full access to the queried information.
- Users with a "Secret" security level go to "DiagnosisCity-HidingValues" which applies a slice restricting values from the "oncology" diagnosis (this constraint was defined in the static model as an SIAR) and then execute a dice operation showing all diagnosis members and only the non empty city members.
- Finally, users with a lower security level ("Confidential" or "Undefined") go to a "DiagnosisCity-ShowingCities" state in which the same slice is applied but in this case, the designer has decided to hide the diagnosis and only show visible city members. Although these users can access city information (a security level of "You" is required), if the designer decides to show all the city members, users could infer information from empty cities which might appear because all their patients have an "oncology" diagnosis.

The users are then split according to their security privileges and they are now shown diagnosis related to cities through the application of different restrictions depending on their privileges and the designer's decisions.

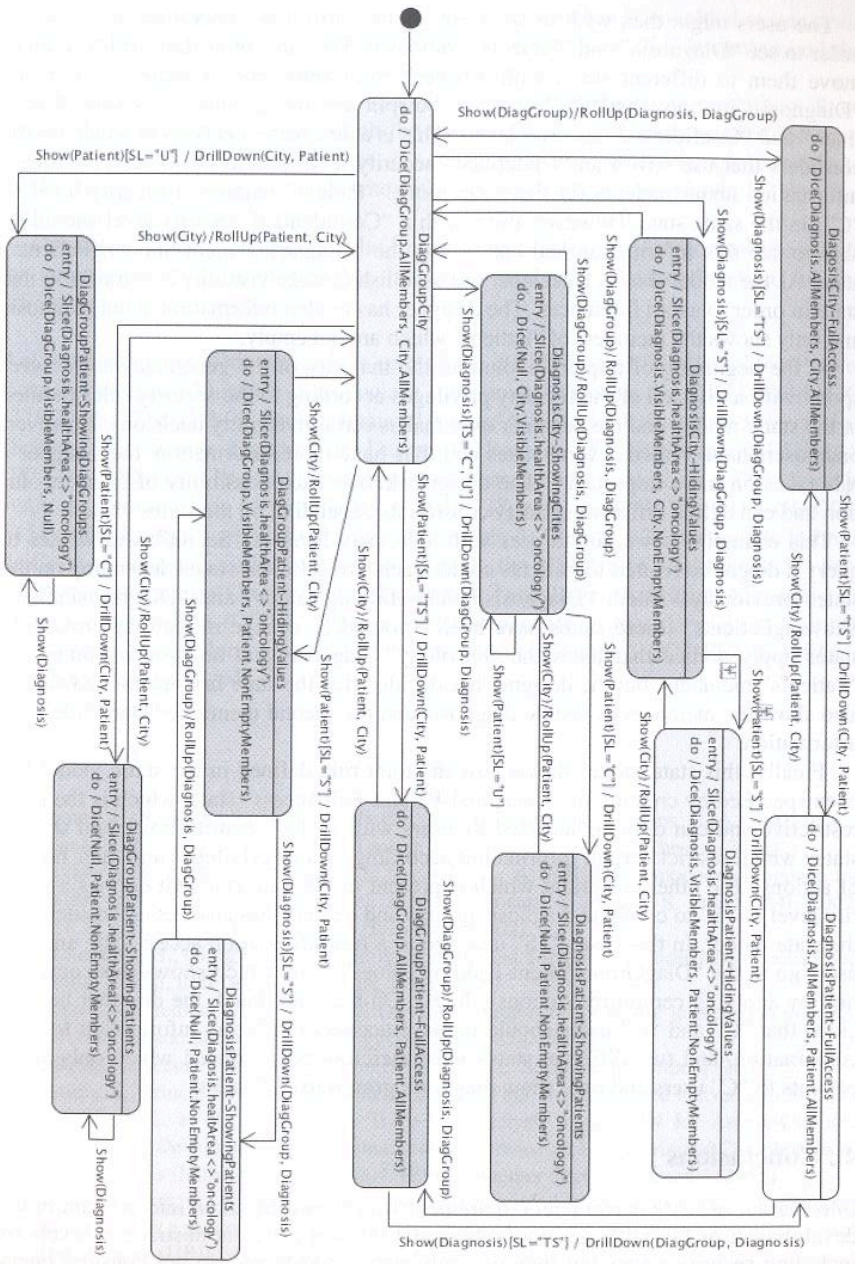

Fig. 4. Secure state model for Diagnosis/Patient

The users might then wish to carry out another drill down operation over cities in order to see "Diagnosis" and "Patients" combined. The same procedure will be used to move them to different states with different constraints. For example, users in the "DiagnosisCity-ShowingCities" state will be split into two groups: users with "Undefined" and "Confidential" security levels. This is a designer's decision in which he/she considers that users with an "Undefined" security level should not be able to access information about patients (in the static model "Patient" requires a security level of "C") in the same state. However, users with a "Confidential" security level should be able to see this information and can access the "DiagnosisPatient-ShowingPatients" state. As the model shows, a designer has established some visibility constraints in this state in order to avoid inferences. The designer has hidden information about diagnosis and only shows the members of "Patient" which are not empty.

At the beginning of a query evolution, the majority of the generated states correspond with a division of the security privileges according to the security rules defined in the static model, and the designer only makes certain visibility decisions. However, once users have visited several states and thus have more information, the designer's role takes on more importance. The designer knows each possibility of query evolution and can define different security constraints depending on the states visited.

This example shows how a user with a security level of "Secret" who wishes to query a diagnosis related to patients could reach two different states depending on the states previously visited: "DiagnosisPatient-HiddingValues" and "DiagnosisPatient-ShowingPatients" (these states have been coloured on orange in the diagram). Both states apply a slice that hides the "oncology" diagnosis and only show non-empty "Patients" members, but the designer has decided that the state first mentioned should also show the members of visible diagnosis and the second mentioned state hides this information.

Finally, this state model shows how the joint rule defined in the static model has been specified by creating the "DiagnosisPatient-FullAccess" state, which is the most restrictive and can only be accessed by users with a "TS" security level, and several states which restrict certain information according to user privileges and their history of actions. The other constraint which is present in the joint rule that defines a security level of "S" to combine diagnosis groups and patients has also been considered in the state model. In this case, "TS" users go to a full information access state and "S" users go to the "DiagGroupPatient-HiddingValues" state, which shows this combination by applying certain restrictions (slides). On the other hand, the designer has decided that "C" and "U" users should be have access to different information for this combination, and two different states have therefore been created which only show patients to "C" users and only show diagnosis groups to "U" users.

4 Conclusions

Information security is a crucial requirement which must be taken into account in the development of DWs. Some proposals model DWs at different abstraction levels by including security issues but they use only static models and do not consider query evolution modelling. Although static security measures have been defined, an attacker could achieve a sequence of OLAP operations which would provide him/her with unauthorized information.

In this paper, a new kind of security rules (joint rules) has been added to our static modelling proposal in order to define sensitive combinations of information and the security privileges needed to query this information.

This approach also fulfils static models with a secure dynamic modelling. State models have been proposed in which the use of several states defines what information has to be provided in each case, by considering aspects that it is impossible to model in a static manner, such as the information that has been shown by the user (previously visited states).

Our future work is focused on the improvement of the session control to include an automatic detection of possible inferences and to recommend actions to the administrator.

Acknowledgments. This research is part of the BUSINESS (PET2008-0136) Project financed by the "Ministerio de Ciencia e Innovación" (Spain), the PEGASO/MAGO (TIN2009-13718-C02-01) Project financed by the "Ministerio de Ciencia e Innovación" (Spain) and "Fondo Europeo de Desarrollo Regional FEDER", the SISTEMAS (PII2I09-0150-3135) Project financed by the "Consejería de Educación y Ciencia de la Junta de Comunidades de Castilla-La Mancha" (Spain), the QUASIMODO (PAC08-0157-0668) Project financed by the "Viceconsejería de Ciencia y Tecnología de la Junta de Comunidades de Castilla-La Mancha" (Spain), the MEDUSAS (IDI-20090557) Project financed by the "Centro para el Desarrollo Tecnológico Industrial. Ministerio de Ciencia e Innovación (CDTI)" (Spain). Partial support from the EU project "Security Engineering for Lifelong Evolvable Systems (Secure Change)" (ICT-FET-231101) is gratefully acknowledged.

References

1. Inmon, H.: Building the Data Warehouse, 3rd edn. John Wiley & Sons, USA (2002)
2. Kimball, R.: The Data Warehouse Toolkit. John Wiley & Sons, Chichester (2002)
3. Lodderstedt, T., Basin, D., Doser, J.: SecureUML: A UML-based modeling language for model-driven security. In: Jézéquel, J.-M., Hussmann, H., Cook, S. (eds.) UML 2002. LNCS, vol. 2460, p. 426. Springer, Heidelberg (2002)
4. Mouratidis, H., Giorgini, P.: Integrating Security and Software Engineering: Advances and Future Vision. Idea Group Publishing, USA (2006)
5. Fernández-Medina, E., et al.: Model-Driven Development for secure information systems. Information and Software Technology 51(5), 809–814 (2009)
6. Jürjens, J.: Secure Systems Development with UML. Springer, Heidelberg (2005)
7. Jurjens, J.: Principles for Secure Systems Design, PhD Thesis, Oxford University (2002)
8. Houmb, S.H., et al.: Cost-Benefit Trade-Off Analysis using BBN for Aspect-Oriented Risk-Driven Development. In: International Conference on Engineering of Complex Computer Systems (ICECCS), pp. 195–204. IEEE Computer Society, Shanghai (2005)
9. Thuraisingham, B., Kantarcioglu, M., Iyer, S.: Extended RBAC-based design and implementation for a secure data warehouse. International Journal of Business Intelligence and Data Mining (IJBIDM) 2(4), 367–382 (2007)
10. Priebe, T., Pernul, G.: A Pragmatic Approach to Conceptual Modeling of OLAP Security. In: Kunii, H.S., Jajodia, S., Sølvberg, A. (eds.) ER 2001. LNCS, vol. 2224, p. 311. Springer, Heidelberg (2001)

11. Fernández-Medina, E., Trujillo, J., Piattini, M.: Model Driven Multidimensional Modeling of Secure Data Warehouses. European Journal of Information Systems 16, 374–389 (2007)
12. Saltor, F., et al.: Building Secure Data Warehouse Schemas from Federated Information Systems. In: Bestougeff, H., Dubois, J.E., Thuraisingham, B. (eds.) Heterogeneous Inf. Exchange and Organizational Hubs, pp. 123–134. Kluwer Academic Publisher, Dordrecht (2002)
13. Trujillo, J., et al.: A UML 2.0 Profile to define Security Requirements for DataWarehouses. Computer Standard and Interfaces 31(5), 969–983 (2009)
14. Trujillo, J., et al.: An Engineering Process for Developing Secure Data Warehouses. Information and Software Technology 51(6) (2009)
15. Fernández-Medina, E., et al.: Access Control and Audit Model for the Multidimensional Modeling of Data Warehouses. Decision Support Systems 42, 1270–1289 (2006)
16. Soler, E., et al.: Building a secure star schema in data warehouses by an extension of the relational package from CWM. Computer Standard and Interfaces 30(6), 341–350 (2008)
17. Blanco, C., et al.: Applying QVT in order to implement Secure Data Warehouses in SQL Server Analysis Services. Journal of Research and Practice in Information Technology 41(2), 135–154 (2009)
18. Fernández-Medina, E., et al.: Developing Secure Data Warehouses with a UML extension. Information Systems 32(6), 826–856 (2007)

A Formal P3P Semantics for Composite Services

Assadarat Khurat[1], Dieter Gollmann[1], and Joerg Abendroth[2]

[1] Hamburg University of Technology,
Harburger Schlossstr. 20, 21079 Hamburg, Germany
assadarat@yahoo.com, diego@tu-harburg.de
[2] Nokia Siemens Networks,
St.Martin-str.53, 81669 Munich
joerg.abendroth@nsn.com

Abstract. As online services are moving from the single service to the composite service paradigm, privacy is becoming an important issue due to the amount of user data being collected and stored. The Platform for Privacy Preferences (P3P) was defined to provide privacy protection by enabling services to express their privacy practices, which in turn helps users decide whether to use the services or not. However, P3P was designed for the single service model, bringing some challenges when employing it with composite services. Moreover the P3P language may lead to misinterpretation by P3P user agents due to its flexibility and may have internal semantic inconsistencies due to a lack of clear semantics. Therefore, we enhance P3P to be able to support composite services, propose a formal semantic for P3P to preserve semantic consistency, and also define combining methods to obtain the privacy policies of composite services.

Keywords: formal semantics, P3P, privacy policy, composite service.

1 Introduction

In the beginning of the online services era, most services were single and independent, employing and developing proprietary technology to serve their customers. Nowadays, there is strong competition in the online market to expand the number of customers. This is an incentive for developing new and better services to better serve user demands. It is then a promising approach to build new services by combining existing services which can reduce development cost and time compared to implementing a new service from scratch. Thus, one can say that the online service world now has changed from single to composite services [12].

In online services, privacy is an important issue due to the large amount of user data collected and stored. Users need some mechanism to secure their data. To protect privacy, users ought first to be aware of what the services will do with their data so that they can decide whether to use the services or not. This facility is provided by the Platform for Privacy Preferences (P3P) [1], a policy language for describing data practices of websites. Comparing these practices with the