Preventive inference control between user profiles in data warehouse

Amina Elouazzani, Salah Ouederrou, Sara Ibn El Ahrache, Nouria Harbi,
Hassan Badir

Preventive inference control between user profiles in data warehouse

Amina ELOUAZZANI*, Salah OUEDERROU*, Sara IBN EL AHRACHE*, Nouria HARBI** and Hassan BADIR*

* LabTIC Laboratory, National School of Applied Sciences Tangier,Morocco

**Eric Laboratory, Lyon II University, France

The detections of inferences between multidimensional queries tend to be more sensitive than data separately. These combinations must be modeled in order to avoid the inference of unauthorized data. However, no work has dealt with inference management in the case of a user who combines two or more permissions. In this paper, we present our approach that focuses on detecting inferences of sensitive data of DW (Data Warehouse) by a user occupying one or more profiles within the company, using the graphical presentation of the permissions and Exploiting associations of source databases.

Keywords: Data warehouse, Inference, Sensitive data, User profiles.

1. INTRODUCTION

In this article, we focus on the field of implicit content whose anchoring is invisible taking into account the available data. We prefer to use the term inference rather than implicit. We define inference as any implicit proposition to extract or deduce prohibited information by combining accessible information. The management of inferences is the subject of several works (Accorsi and Müller, 2013) (Chen and He, 2010); however the detection of inferences by the combination of data has not been of much interest to researchers despite its importance for DWs that use multiple data sources. According to the work of (Blanco, Fernández-Medina, Trujillo, and Jurjens, 2010), queries that require the combination of several multidimensional elements tend to be more sensitive than data separately. These combinations must be modeled in order to avoid the inference of unauthorized data. In order to have a proactive system, our proposition allows us to produce knowledge about inferences between permissions combined by the same user; Knowing that the latter can have one or several roles within the company. Each role contains permissions. A user can infer sensitive data by combining the permissions granted. In this paper, we present our approach that focuses on detecting inferences of sensitive DW data by a user occupying one or more profiles within the company, using the graphical presentation of the permissions and exploiting associations of source databases.

2. STATE OF THE ART AND SYNTHESIS

2.1 Existing work

A number of methods have been proposed to deal inference attacks. In (Sellami, Hacid, and Gammoudi, 2015), the authors proposed a methodology that allows controlling the access to a data integration system. In such systems, a mediator is defined. This mediator aims at providing a unique entry point to several heterogeneous sources. The methodology allows dealing with direct access and indirect access. The methodology includes three main phases: (1) the generation of a global schema, global functional dependencies and a global policy from local sources and underlying policies, (2) disclosure transaction discovery by exploiting the semantics of data dependencies. (3) Proceed to policy reconfiguration to avoid security breaches. The work (Albertini, Alberto, Carminati, and Ferrari, 2017), proposes a mechanism to avoid users to infer additional

non-authorized data by linking implicitly authorized attributes returned by a rewritten query. The key idea is to exploit a non-harmful dependency to extend the access rights that users have on attributes in determinant part over attributes in dependent.

Management of inferences has inspired and is still inspired today, of the work carried out in the field of DWs or databases in general. We find in the literature the work of (Triki, BEN ABDALLAH, BOUSSAID, and HARBI, 2013) which proposes a model for securing multidimensional data against inferences in the conceptual phase, this approach makes it possible to detect the precise inference where the values of the inferred data are accurate. And the partial inference where the values of the data are partially disclosed, ie the user can deduce an idea about the value of the data.

This approach consists of identifying of the sensitive elements to be protected. Then, the owner constructs a graph of inferences from the class diagram, specifying the elements that present precise or partial. And the last step is to present of the DW with the UML annotations by highlighting the two types of inferences.

On the other hand, the work of (Blanco et al., 2010) Propose a state-of-the-business diagram approach to detect inferences at the design level. This proposal focuses on sensitive queries and their evolutions. It is presented in the form of a 3-state OLAP security model. The Static model which presents the UML profile specific to DWs (Fernandez-Medina, Trujillo, Villarroel, and Piattini, 2006), adding a new kind of rule. Dynamic model which aims to enrich the static model, by dealing with the evolutions of the combinations, through the application of OLAP operations. And sessions control this step which makes more use of users' sessions to analyze them by checking each event to detect any possibility of inference. This work makes it possible to study the evolutions of a request but it does not take into account the inference of the data from the accessible data. It indicates that the combination of several permissions may be more sensitive, which is approved in the work of (Sweeney, 2002).

The work of (Sweeney, 2002) Describes a real case of data inference, by demonstrating the identification of sensitive data based on the cross-linking of an insurance group's data, assuming that they are anonymous, and a voter registration list, which allowed him to detect the name of former governor «William Weld» and his medical records, linking the shared attributes. This work remains a real case of a sensible combination of inference, since it does not propose an effective solution.

We also find in the literature the work of (Chen and He, 2010) which proposes an inference detection model which makes it possible to acquire knowledge such as dependencies between attributes within the same entity and even between entities, rules and constraints, in order to represent all the possible relations between the attributes of the databases. These inference graphs will be used to compare the new query requested with the knowledge base acquired without taking into account the profiles and permissions of a user.

We also find the work of (Accorsi and Müller, 2013) which proposes rules of inference known by the inference engine, without mentioning how to specify them. The proposed diagram shows an inference detection process that consists of a composite policy in which the user composes the policy and privacy rules. Then the inference closure calculation in turn computes all possible inference closures of the input policy based on an algorithm that represents the necessary steps. And the kernel tests for each non-kernel element if it is obtained from a kernel element. This work is limited since the rules for detecting inferences are not specified.

2.2   Synthesis

The protection of DWs against illegal access has been undoubtedly sensed for several years (Fernandez-Medina et al., 2006), (Soler, Trujillo, Fernandez-Medina, and Piattini, 2008), (Trujillo, Soler, Blanco, and Fernandez-Medina, 2009), (Arora and Kumar, 2016), (Eavis and Althamimi, 2012). However, the notion of inference has been cited in several works as an essential element for ensuring confidentiality, and the mastery of which is crucial.. Following the review of existing work, we noted the following:

—We find works that deal with inference attacks in relational databases (Sellami et al., 2015), (Albertini et al., 2017). these works are based on functional dependencies in order to detect inferences. however, these proposals can not protect a DW that uses multiple data sources.
—Although permissions present the primary focus to ensure the confidentiality of DW access, the absence of a standard that manages the accuracy of these permissions can cause inconsistencies and inferences. In this sense, the work of (Saltor, Oliva, Abello, and Samos, 2002) has proposed the use of the authorization scheme defined for federated databases without any modification to build a secure DW, and (Rosenthal and Sciore, 2000) proposed the rewriting of the queries in order to verify that the latter comply with the restrictions defined at the source level.
—Nevertheless, despite the high risks of inferences, it is not sufficiently taken into account in the conceptual phase.
—The conflict of interest between permissions granted to a user has not been processed.
—No work offers a convivial method to detect sensitive combinations that can cause inferences.
—No work offers a method that ensures the consistency of a user's permissions according to his profile.

In this sense, we have already proposed a method for classifying data according to their level of sensitivity (Ouazzani, Rhazlane, N.Harbi, and H.Badir, 2016), in order to identify the potentially data subject of inferences. In this paper, we propose a visual method for the automatic detection of sensitive combinations among the permissions of a user, which can cause an inference of the sensitive data. This method consists of five rules, and it takes into account the profiles of a user in order to ensure the consistency of the permissions granted to a single user. This has an advantage over existing work.

3.   THE DETECTION OF INFERENCES BY COMBINING SEVERAL PROFILES

3.1   Basic Concept

In order to allow for an automatic extraction of inferences from the permissions allowed, we propose a visual computer model with rules to be checked based on the graphical representation of the profiles granted to a user and the links between the data using the source class diagram. This approach is the continuation of the work of (Triki et al., 2013) which proposes a method for detecting precise and partial inferences, but our proposal is to detect sensitive combinations.

Knowing that a user can have one or more roles within the company, the user accesses the DW with one or more profiles. The purpose of our inference detection system is to detect if a user can indirectly deduce unauthorized information by using two or more permissions from one or more different profiles.

3.2   Description of the Proposed Architecture

The proposed global architecture (Figure 1) presents our user-based access control model, which consists of three modules. The first module deals with the dynamic classification of sensitivity levels of warehouse data, based on user profiles (Ouazzani et al., 2016). The second module of our architecture, which we will detail in this part, focuses on the visual detection of inferences

through combining several permissions of the same user, to eventually send the detected sensitive combinations to the data owner.
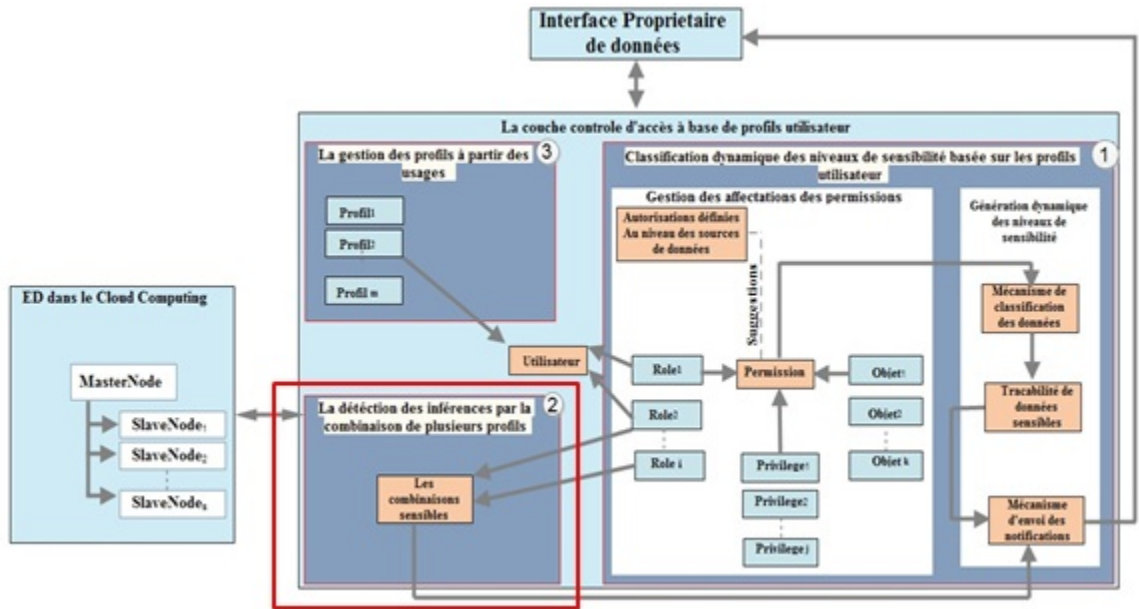

Figure 1. The proposed global architecture

Figure 2 shows the detailed architecture of our inference detection module, which allows to analyze the permissions of each profile, in order to detect the sensitive combinations. The module is based on two inputs which are:
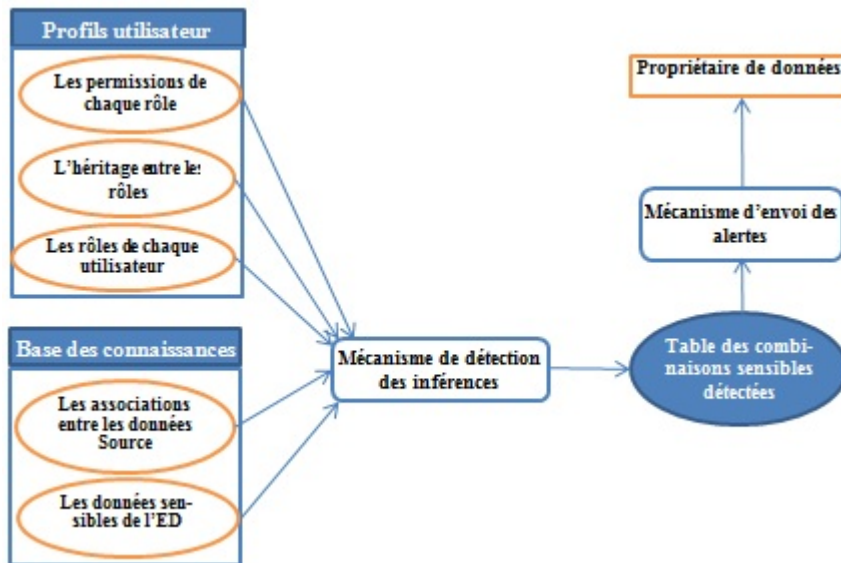

Figure 2. Detection of inferences by the combination of several profiles

—User profiles: present users who endorse roles assigned to them by the data owner. These roles are organized in hierarchy, and they have permissions that cannot be granted directly to users.

—The knowledge base: these are the sensitive data of the DW to be protected against inferences, and the associations between the data according to the class diagram of the source database.

3.3 Inference detection mechanism

The analysis of the permissions granted to a user according to his roles is an important process for the data owner and for the company. It is used to limit the risks and improve the quality of permissions assignment. It allows to identify the anomalies in the process of the assignment of the permissions in order to correct them. It can be considered as a means of prevention for the data owner by detecting the different inference risks from the permissions granted by using the class diagram of the data sources.

3.3.1 *Meta-Model architecture.* The proposed meta-model presents an extension of the standard CWM (Common Warehouse Meta-model). It describes the DW elements presented by the class «ObjectDW». An «ObjectDW» can be a table (Fact, Dimension, Base), a column, or a value of a column. The proposed meta-model also describes the source databases of the DW presented by the class «ObjectSource». An «ObjectSource» can be a table, attribute, or value of an attribute, and it can relate to other objects. At first we only work with class diagram source databases. It contains also five classes (PermissionDW, SourcePermission, AutomaticSecurityLevels, Threshold, Traceability) considered the core of our contribution (Ouazzani et al., 2016).
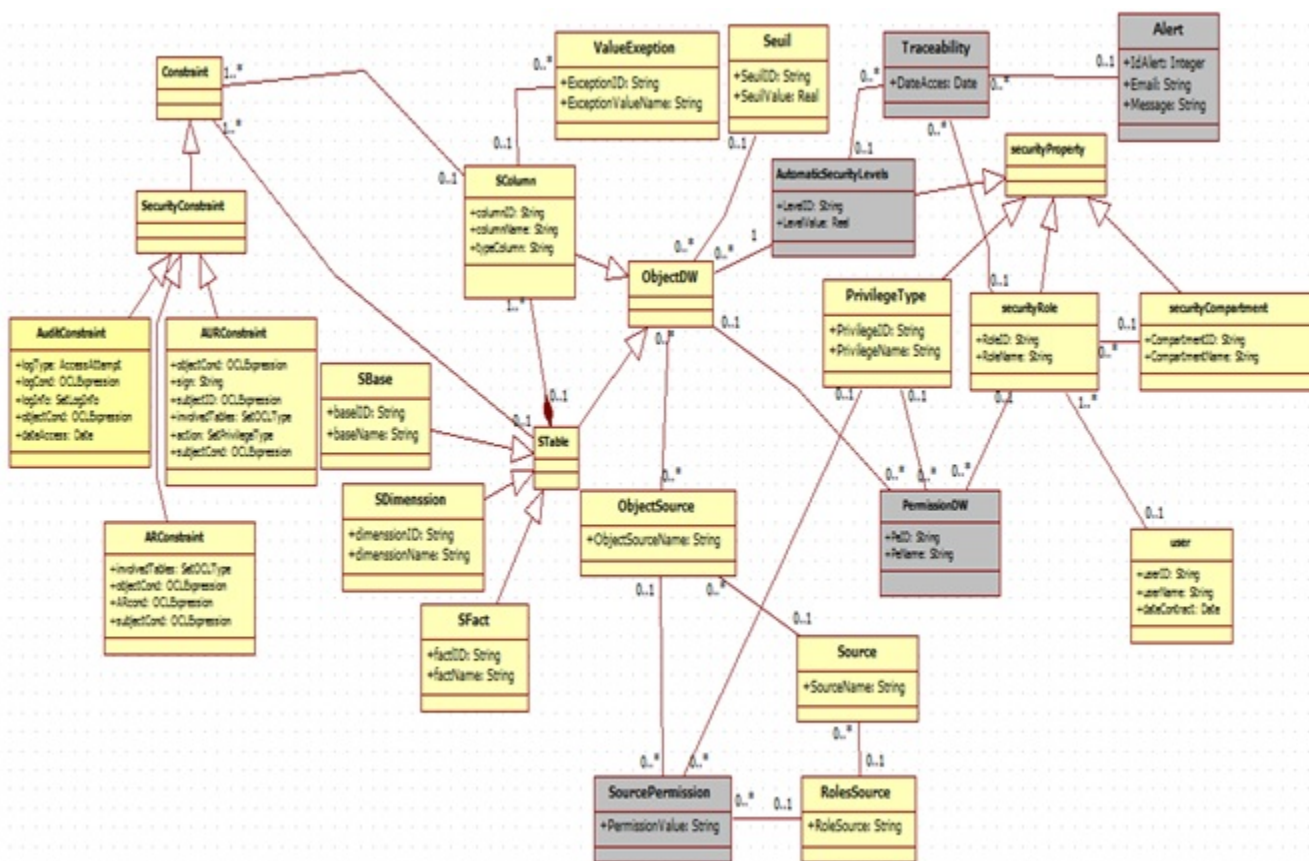

Figure 3. The proposed architecture

3.3.2 *User Profiles Presentation using Graphs.* In order to model the profiles and the permissions of each user, we rely on the notion of graphs that we find in the literature (Triki et al., 2013), (De Capitani di Vimercati, 2008). A permission p is a rule of the form $[O, R] \Rightarrow S$ which stipulates

that a user S who occupies the R role has the right to access the object O.

In our model user profiles are presented by graphs where the nodes are the elements of a profile (user, role, objects). An object of the DW can be a column, a table, or a value in a column. An arrow (Table 1) between a role and the objects represents the access rights of a role to all the objects of the DW. An arrow between two objects represents an association according to the class diagram of the source database. On the other hand, an arrow between two roles represents an inheritance. The nodes of the first level of the graph present the users, the nodes of the second level of the graph present the roles of a user and the other levels contain the objects of the DW. Colored nodes are sensitive DW objects.

Elément	Signification
U	A node representing a user
R	A node representing o role
O	A node Representing a non-sensitive object of a DW
O	A node Representing a sensitive object of a DW
→	—A continuous line arc between a role and objects of a DW presents the permissions of a role. —A continuous line arc between two objects of a DW indicates a source association between two objects allowed to the user. —A continuous line arc between two roles indicates an inheritance between two roles.
--→	A dotted arc indicating an association that causes an inference, from an allowed object to another not allowed object to the user.

Table I: MEANING OF GRAPHIC ELEMENTS

Example: Let's take a simple example to illustrate the construction of a graph. Let Alice be a user who occupies two roles within the company Medical Secretariat Analyzer, Manager Analyzer.

Permission P1 : $[Patient, MedicalSecretariatAnalyzer] \Rightarrow Alice$ means that the user Alice who occupies the role «Medical Secretariat Analyzer»has the right to view the sensitive object «Patient».
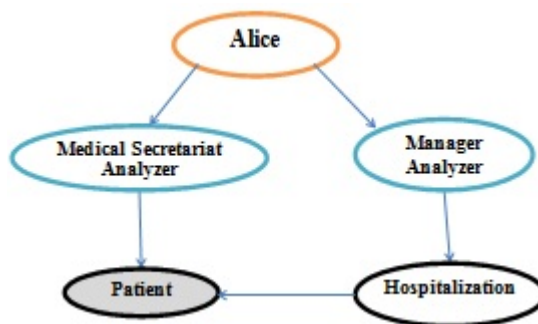
Figure 4. Example of a graph

Permission P2 : $[Hospitalization, ManagerAnalyzer] \Rightarrow Alice$ means that the user Alice who occupies the role of a «Manager Analyzer» has the right to consult the non-sensitive object «Hospitalization».

Figure 4 schematizes the permissions of each role occupied by user Alice. Only the «Patient» object of the graph is colored, because it presents a sensitive object. The arrow between the «Hospitalization» and «Patient» objects represents an association between them according to the source class diagram, indicating that the «Hospitalization» object contains the primary key of the «Patient» object which is the CIN (National Code of Identity), and consequently it is an inference.

3.3.3 *Sensitive Data Definition* . In general, sensitive data are data classified by the first module of our global architecture (Figure 1), whose sensitivity level is higher than the threshold set by the data owner (Ouazzani et al., 2016). The data to be protected against inferences are the unauthorized sensitive data to a user which can be deduced by combining permissions of profiles granted to a user.

3.3.4 *Rules of inference between user permissions.* In order to check the consistency of the permissions granted to a user according to his profiles, without risk of inferring unauthorized sensitive data, we have defined five rules for defining the sensitive combinations:

a) **By unauthorized intermediary:** It is a rule that makes it possible to detect an inference by the combination of the allowed data (FIG. 5), using an unauthorized intermediary O3 which may be sensitive or non-sensitive containing a common field between two permissible permissions.
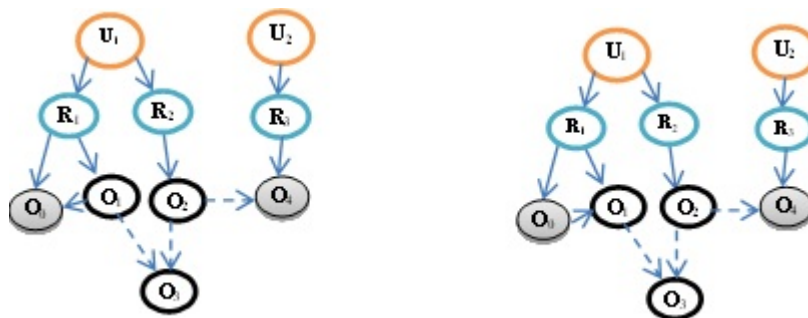

Figure 5. Inference rule by unauthorized intermediary

The combination (O1, O2) is a sensitive combination. As a consequence, it will be necessary to verify the existence of a link between these objects at the DW level.

b) **Through a direct passage:** This is a rule that allows an inference to be detected by the combination of multiple access permissions. According to figure 6, and in order to deduce information on the unauthorized sensitive object O3, the user U1 can use a query on the object O2 which has a passage between the sensitive objects allowed to the user according to the role R1 And the sensitive object O3.
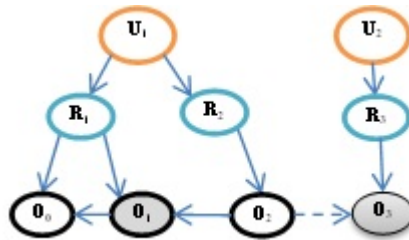

Figure 6. Rule of inference by a direct passage

In this case and according to this graphic presentation, the sensitive combination consists of the objects (O1, O2), and consequently it will be necessary to verify the existence of a link between these objects at the level of the DW.

c) **By an authorized intermediary:** It is a rule that allows to detect an inference by using a common field between two permissions assign to a user according to one or more profiles, using queries with the common field. According to figure 7, the sensible combination consists of the object O1 and the intermediate object O3.
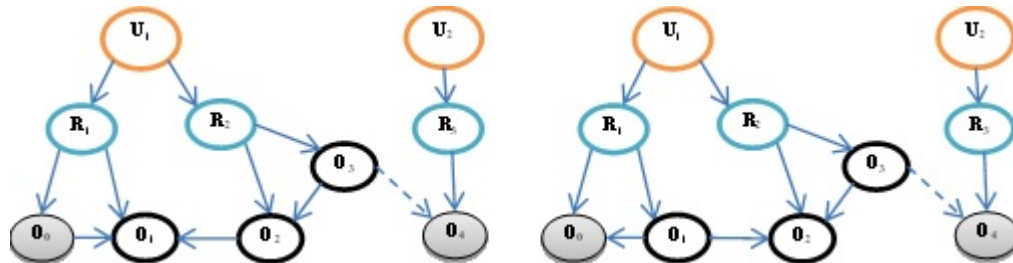

Figure 7. Inference rule by an authorized intermediary

As a consequence, it will be necessary to verify the existence of a link between these objects at the DW level.

d) **Through an invisible passage:** The rule of inference by an invisible passage according to the graphic presentation (figure 8) makes it possible to determine a sensitive combination without any visible intermediate. By opening a session with role R2, user U1 can execute a query with a criterion on the common field between O2 and O3. Next, the user has the option of logging in with R1 to execute another query on object O1, with a criterion containing the result of the first query. This creates an inference of the O3 object that is not allowed.
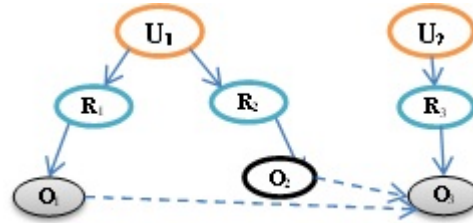
Figure 8. Inference rule by invisible passage

Therefore, we can deduce that the combination (O1, O2) is a sensible combination. As a consequence, it will be necessary to verify the existence of a link between these objects at the DW level.

e) **By inheritance:** This rule is used to detect inheritance inference. According to figure 9, role R3 has the right to read the object O3, and it inherits the permissions of the role R2. The latter has permission to access object O2, and according to the source class diagram, object O2 is in association with object O1. And consequently user U2 can execute queries with the role R3, in order to use the result in another query with the R2 to deduce information on the unauthorized sensitive object O1. We can see that the combination (O2, O3) is a sensible combination. As a consequence, it will be necessary to verify the existence of a link between these objects at the DW level.
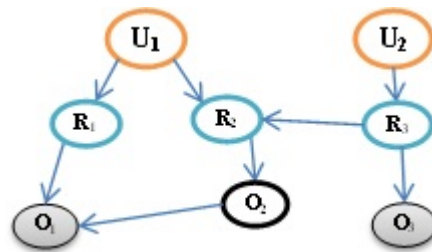

Figure 9. Inference rule by inheritance

Detecting an inference according to the proposed rules, using the source class diagram, involves checking a link/association between rule objects according to the schema of the DW to detect unauthorized inferences.

3.3.5 *Checking rules.* Detecting an inference according to the proposed rules, using the source class diagram, involves checking a link/association between rule objects according to the schema of the DW to detect unauthorized inferences.

4. INTERPRETATION EXAMPLE

Among the sensitive and secret data of our DW, there is the type of disease of a given patient, as well as information on the invoices paid. This data should not be inferred by a combination of authorized data. In this example we will detect the sensitive combinations that can infer this sensitive data, using the proposed rules. We will use the DW of Test (Figure 10), with the hierarchy of roles shown in Figure 11.
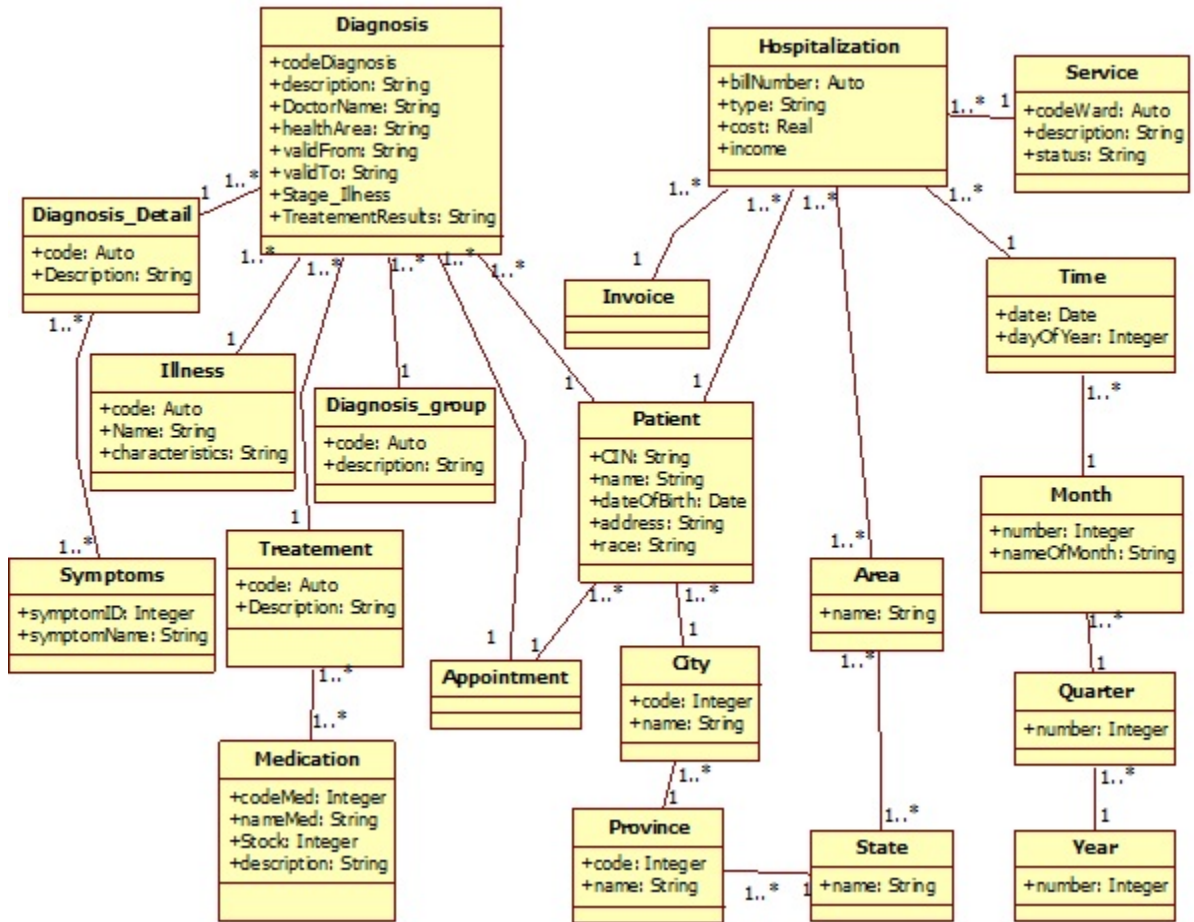
Figure 10. Data Warehouse TEST

The class diagram of the source databases is shown in Figure 12, which shows the associations between the data we are going to use in order to detect the possible inferences by combining the data.
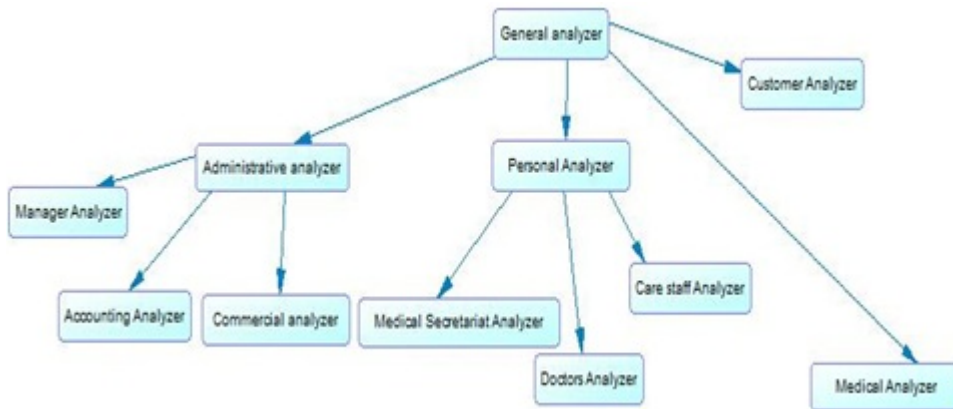

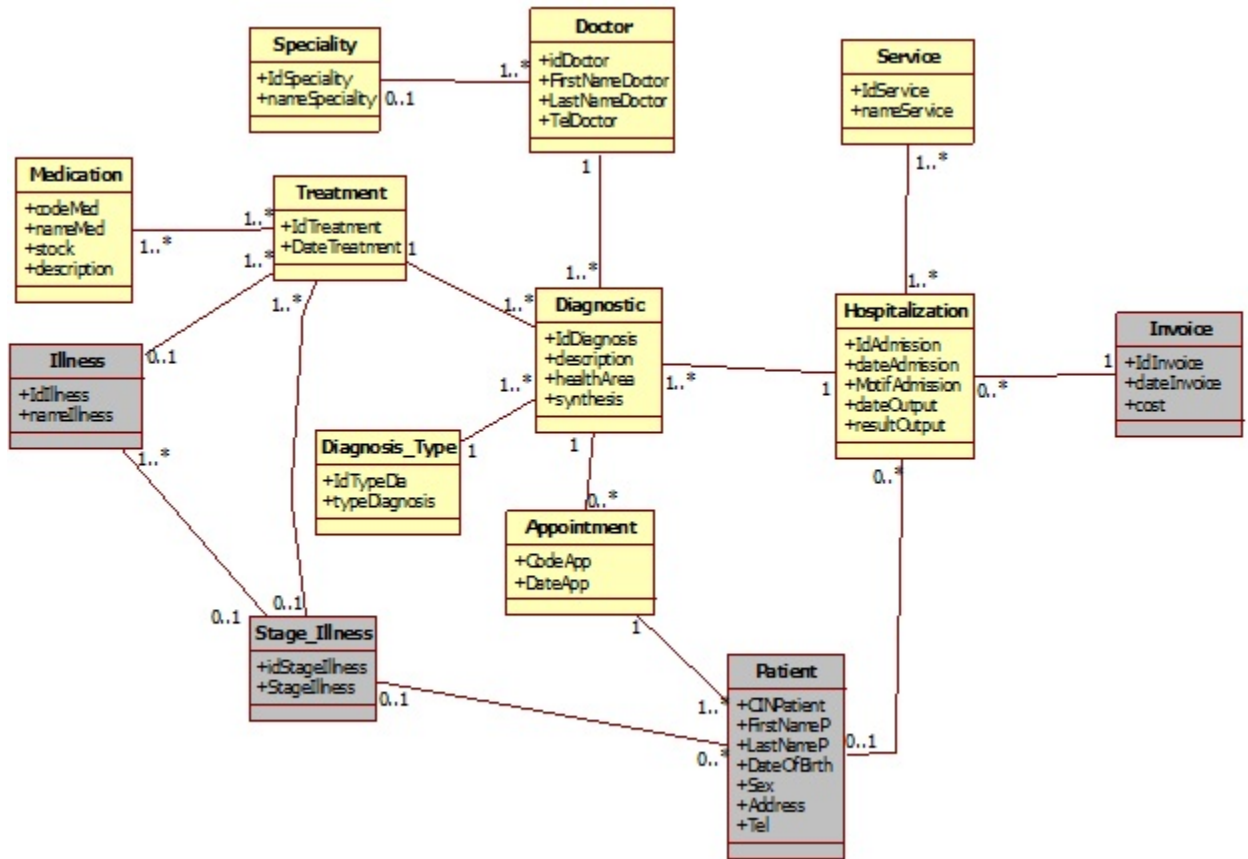
Figure 11. Example of a role hierarchy

Figure 12. Source class diagram
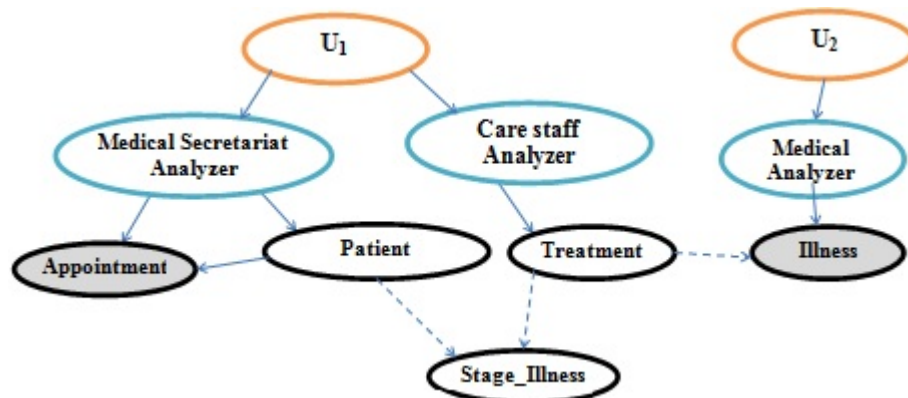
4.1 By unauthorized intermediary

Figure 13. Example of inference by unauthorized intermediary

According to Figure 13, our user U1 occupies a role «Care staff Analyzer» and a role «Medical Secretariat Analyzer». According to the source class diagram, and in order to infer information about a patient's disease, user U1 can log in with the role «Care staff Analyzer» and execute

a first query to know the patient's appointments since, according to our Source class diagram (Figure 12), the primary key of the «Appointment»table is the foreign key of the Patient table. Then, he can execute another query on the processing table using the result of the first query in order to deduce an idea about the patient's disease since the Primary key of the «Illness»table is the foreign key of the «Treatment»table. And consequently the combination (Appointment, Treatment) is a sensible combination. After verifying the existence of a link between these objects at the DW level, we found that this rule is not true.

4.2   By an authorized intermediary

In this example (Figure 14), and according to the source class diagram, user U1 can use the authorized object «Hospitalization»as a passage in order to infer information about the patient's sensitive invoice object.
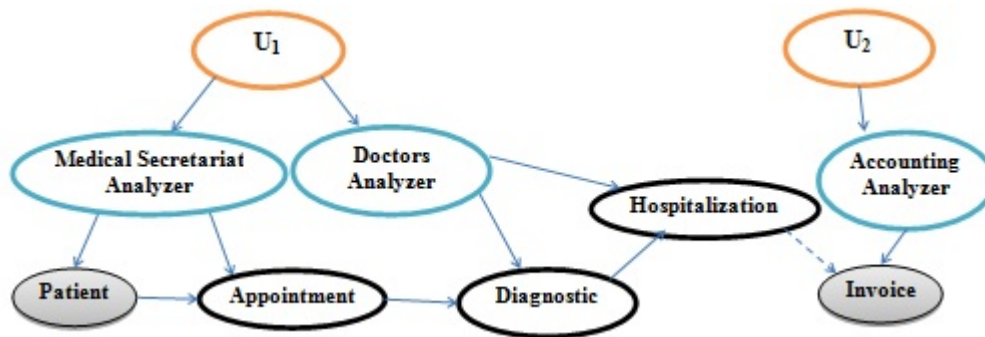

Figure 14. Example of an inference by an authorized intermediary

   User U1 has two profiles with two roles «Medical Secretariat Analyzer»and «Doctors Analyzer». With the role «Medical Secretariat Analyzer», user U1 can execute a query to know the appointments of a given patient. The result of this query can be used as a criterion in a query by logging in with the «Doctors Analyzer»role in order to deduce an idea on a patient's invoice since the primary key of the «Invoice»table is a foreign key in the «Hospitalization»table.
And therefore the combination (Hospitalization, Appointment) is a combination at risk. According to the DW diagram, we can notice the existence of a link between these objects, so there is an inference of type «authorized intermediary».
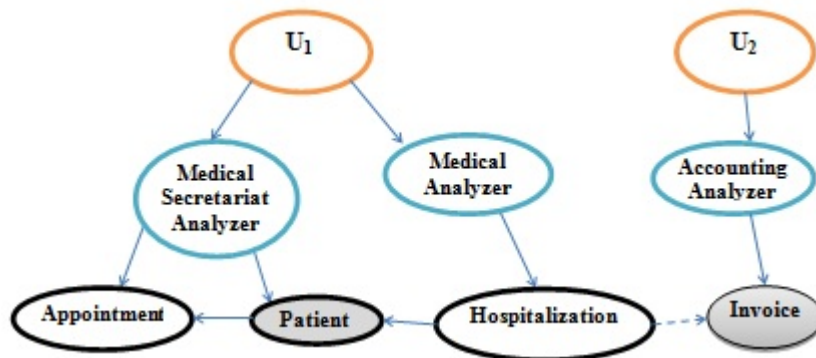
4.3   By a direct Passage

Figure 15. Example of an inference by a direct passage

User U1 who combines two profiles with the two roles «Medical Secretariat Analyzer» and «Medical Analyzer» (Figure 15), can deduce an idea about the sensitive object «Invoice», using the object «Hospitalization» as a direct path, since it contains the primary key of the «Patient» table and the primary key of the «Hospitalization» table as foreign keys. And accordingly, in relation to the source class diagram, the combination (Hospitalization, Patient) is a sensitive combination. And after checking the DW schema, we can see that there is a link between these objects, so there is a «Direct Passage» type inference.

4.4   Through an invisible passage

This graphic presentation (figure 16) makes it possible to visualize, according to the source class diagram, an invisible inference case that can be performed by the user U1 who combines the two profiles of the two roles «Customer Analyzer» and «Doctors Analyzer».
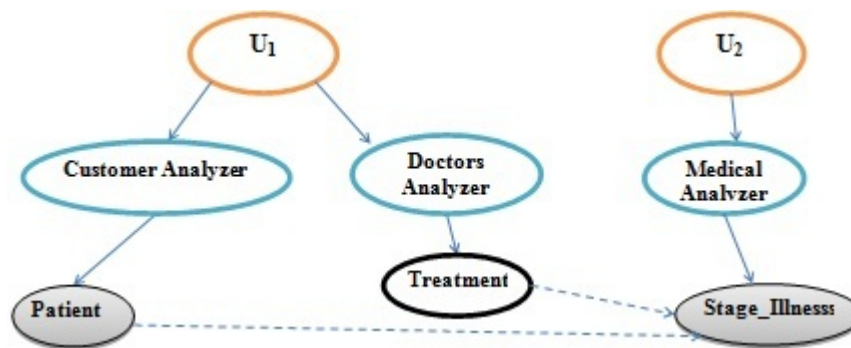

Figure 16. Example of an invisible passage inference

Knowing that the primary key of the «Stage_Illness» table exists in the «Patient» and «Treatment» tables, this user can deduce an idea about the unauthorized sensitive data «Stage_Illness» which is the condition of a patient's disease, as well as the treatment prescribed for the latter. So the combination (Patient, Treatment) is sensitive. After verifying the existence of a link between these objects at the DW level, we found that this rule is not true.

4.5   By inheritance

User U2 is the «Medical Analyzer» who inherits the «Doctors Analyzer» role. According to the source class diagram, the «Treatment» table contains the primary key of the Patient table, user U2 can execute a query that groups together both the «Treatment» and «Stage_Illness» tables in order to deduce information about the Patients and their state of health. After checking the DW schema, we can see that there is a link between these objects, so there is an inheritance inference.
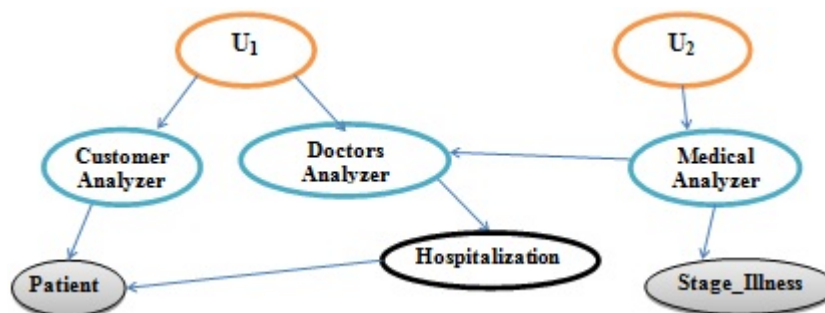

Figure 17. Example of an inference by an inheritance

5. RULES IMPLEMENTATION

5.1 Tools and development environment

For the implementation of our contribution and the realization of the experiments, we used a machine DELL PRECISION T1700 under Windows 7 professional 64-bit. This machine has a 3.40 Ghz Intel Core i7-4770 processor and 8 GB of RAM. We used the following software tools:

—The Java language with the Eclipse development environment. This choice was motivated by the advantages offered by this language in terms of portability, robustness and the availability of many libraries.

—The MySQL relational DBMS to design our Meta Model. This choice was mainly argued by its simplicity of use and its interfaces to perform various operations.

In order to show the applicability and evaluate the advantages and limitations of our inference detection approach, we developed an application with the JAVA language by implementing the example of section 4 with the DW test (figure 10), the hierarchy of roles (Figure 1), and the source database (Figure 12) that are regrouped in our Meta Model and implemented under the WampServer development platform.

We created 5 users and 11 roles. Each role has 2 to 8 permissions. A profile gathers information about the user and his role with his permissions. A user can have one or more profiles according to the occupied roles. In this example, we created the profile «Salma—It».

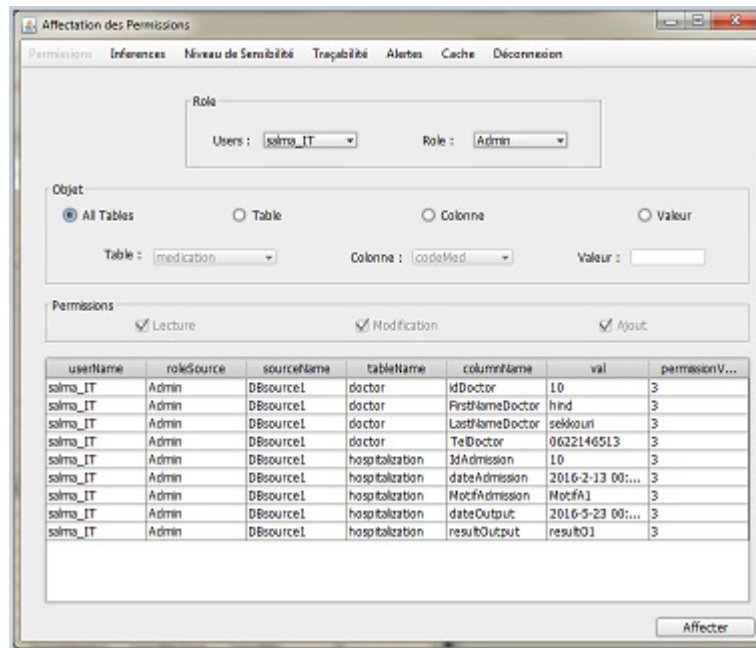### 5.2 Interface for assigning permissions to a role

Figure 18. Interface for assigning permissions to a role

Assigning permissions is a task that only the administrator takes care of. In this interface we choose the role, the program displays the information about this user as well as the objects authorized at the source databases, which we have already treated in (Ouazzani et al., 2016), in order to help the administrator define permissions. In this example, we assigned 8 permissions to the «Medical Analyzer»role.

This interface makes it possible to specify the type of object (table / column / value), and the privileges to be authorized.

5.3 Inference detection interfaces

the interfaces presented in this section displays the profiles assigned to the selected user. Then the administrator can verify the existence of an inference according to the 5 rules proposed in this article.

In the interface 19, the owner has chosen to check the coherence of the profiles occupied by the user user1. Our system has detected an inference through the invisible object unauthorized stage_Illness using the objects Patient, Treatment in order to infer the unauthorized sensitive object Illness.

Figure 19. Interface of the example of the inference rule « By unauthorized intermediary »

in the interface 20, the owner has chosen to check the coherence of the profiles occupied by the user user3 which are "analyzer medical secretary "Analyzer of the doctors". Our system detected an inference by the authorized object "Hospitalisation"using the Diagnostic, Appointment and Patient objects, in order to infer the unauthorized sensitive object Invoice.
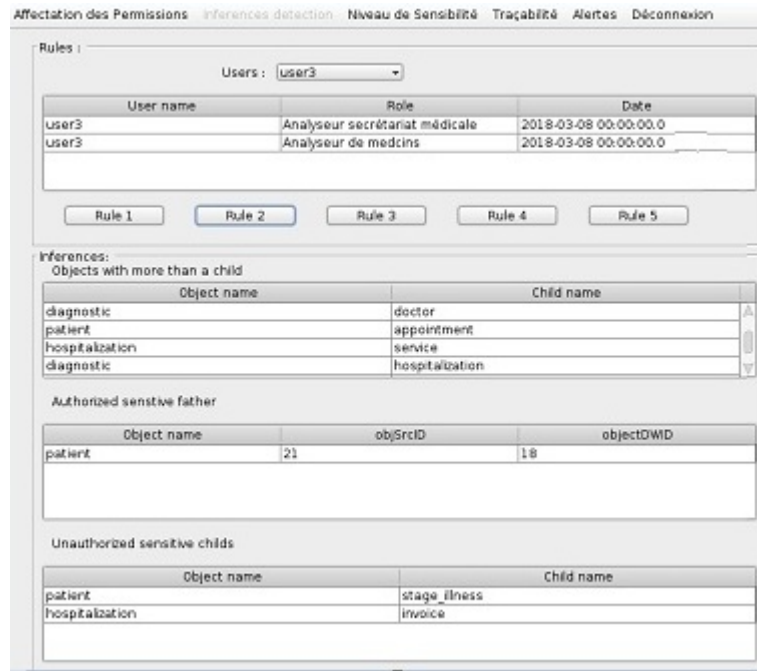
Figure 20. Interface of the example of the inference rule « By an authorized intermediary »

in the interface 21, the owner has chosen to check the coherence of the profiles occupied by the user user5 which are "Medical Secretary Analyzer "Medical Analyzer". Our system has detected an inference by the authorized object Hospitalization that presents the intermidiary between the sensitive object authorized patient and the sensitive object unauthorized invoice.
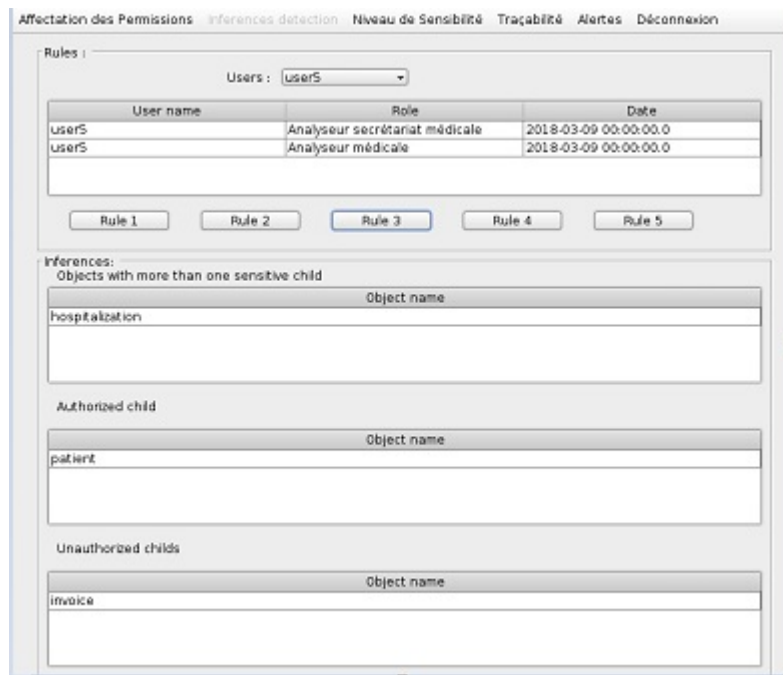

Figure 21. Interface of the example of the inference rule « By a direct Passage »

in the interface 22, the owner has chosen to check the coherence of the profiles occupied by the user user7 which are "Client Analyzer "Doctors Analyzer". Our system detected an inference by a non-visible passage between the unauthorized sensitive object Illness_stage and the authorized objects Treatment and Patient.

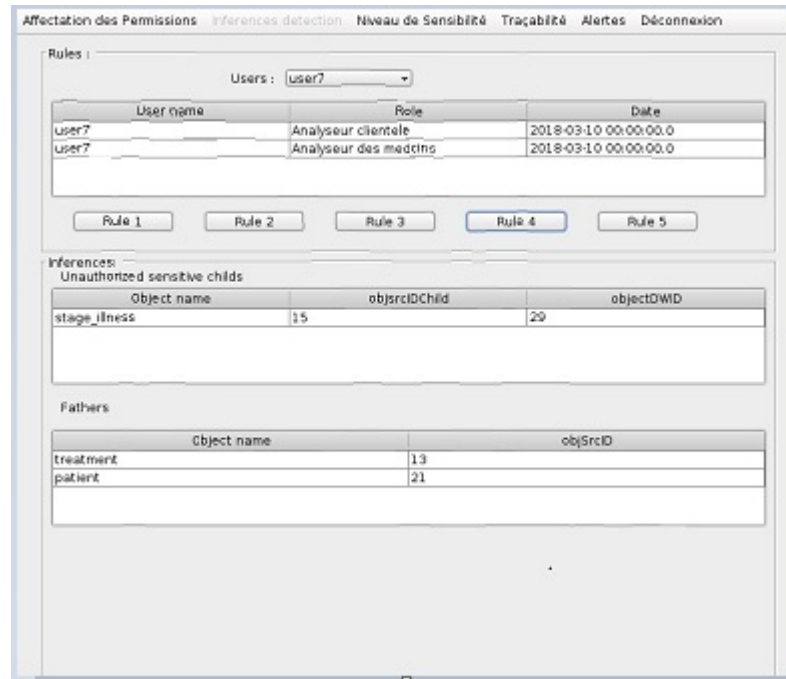Figure 22. Interface of the example of the inference rule « Through an invisible passage »

in the interface 23, the owner has chosen to check the coherence of the profiles occupied by the user user10 which are "Medical Analyzer "Doctors Analyzer". Our system has detected an inheritance inference between the objects authorized Illness_stage, Hospitalization and the unauthorized and sensitive object Patient.
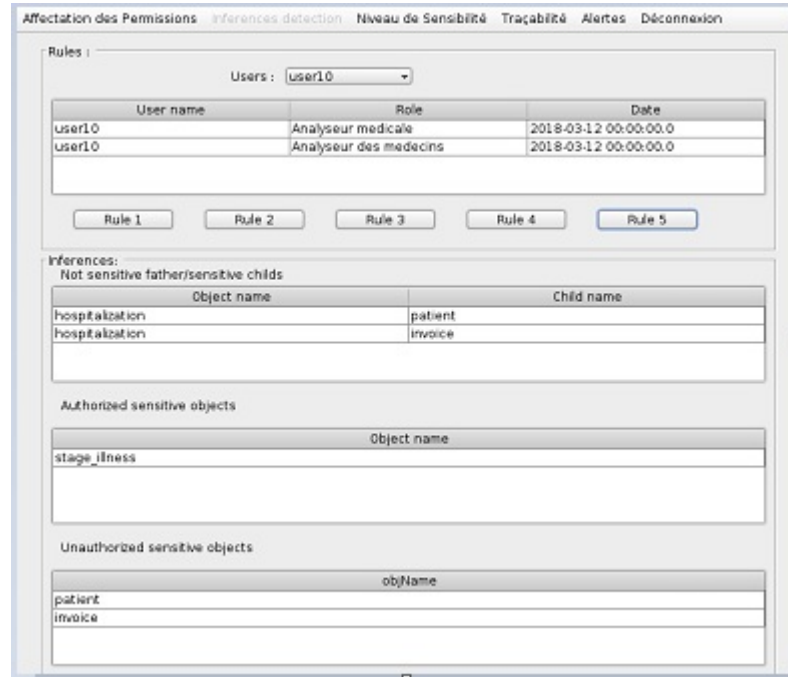
Figure 23. Interface of the example of the inference rule « By inheritance »

Overall the results are encouraging and effective in detecting inferences according to the proposed rules. However additional efforts deserve to be invested in terms of the feeding DW, and the response time of our application which is long since it checks each permission, with all the permissions granted to the user.

6.   THE EFFECTIVENESS OF THE PROPOSED APPROACH

The proposed approach presents five rules for detecting inferences between one or more permissions assigned to a user according to one or more roles. The implemented application allows the data owner to test the existence of the inferences once the roles are assigned to the user. The strong point of these rules is that they detect combinations of sensitive permissions and not a simple inference. However, our approach is just based on source class diagrams, so it does not work in the case of a data warehouse that has several types of sources.

7.   CONCLUSION AND PERSPECTIVES

In this article, we defined the problem of detection of inferences by the combination of permissions allowed. This problem has not been of interest to researchers, despite the importance of checking the consistency of permissions according to one or more profiles assigned to a single user. To this end, we proposed rules to detect sensitive permissions that can infer unauthorized sensitive data. This can help the data owner to properly control the permissions granted. With our example of interpretation, we presented the rules of detection of the inferences proposed in a real case in order to explain them well. Among our perspectives, we intend to look for and improve other rules, allowing to detect inferences between several profiles of a single user of the DW.

Литература

ACCORSI, R. AND MÜLLER, G. 2013. Preventive inference control in data-centric business models. *Security and Privacy Workshops (SPW). IEEE*, 28–33.

ALBERTINI, ALBERTO, D., CARMINATI, B., AND FERRARI, E. 2017. An extended access control mechanism exploiting data dependencies. *International Journal of Information Security*, 75–89.

ARORA, D. AND KUMAR, U. 2016. Protecting sensitive warehouse data through uml based modeling. *Proceedings of the International Conference on Informatics and Analytics*, 31.

BLANCO, C., FERNANDEZ-MEDINA, E., TRUJILLO, J., AND JURJENS, J. 2010. Towards the secure modelling of olap users behaviour.

CHEN, D. AND HE, Y. 2010. A study on secure data storage strategy in cloud computing. *Journal of Convergence Information Technology*.

DE CAPITANI DI VIMERCATI, S. 2008. Assessing query privileges via safe and efficient permission composition. *Proceedings of the 15th ACM conference on Computer and communications secur*.

EAVIS, T. AND ALTHAMIMI, A. 2012. Olap authentication and authorization via queryre-writing. *The Fourth International Conference on Advances in Databases, Knowledge, and Data Applications*, 130–139.

FERNANDEZ-MEDINA, E., TRUJILLO, J., VILLARROEL, R., AND PIATTINI, M. 2006. Access control and audit model for the multidimensional modeling of dws. *Decision Support Systems*, 1270–1289.

OUAZZANI, A., RHAZLANE, S., N.HARBI, AND H.BADIR. 2016. Dynamic management of data warehouse security levels based on user profiles. *Information Science and Technology (CiSt),4th IEEE International Colloquium*, 59–64.

ROSENTHAL, A. AND SCIORE, S. 2000. View security as the basis for data warehouse security.

SALTOR, F., OLIVA, M., ABELLO, A., AND SAMOS, J. 2002. Building secure data warehouse schemas from federated information systems.

SELLAMI, M., HACID, M. S., AND GAMMOUDI, M. M. 2015. Inference control in data integration systems. *OTM Confederated International Conferences"On the Move to Meaningful Internet Systems*.

SOLER, E., TRUJILLO, J., FERNANDEZ-MEDINA, E., AND PIATTINI, M. 2008. Building a secure star schema in data warehouses by an extension of the relational package from cwm. *Computer Standards and Interfaces 30*, 341–350.

SWEENEY, L. 2002. k-anonymity: A model for protecting privacy.

TRIKI, S. ., BEN ABDALLAH, H., BOUSSAID, O., AND HARBI, N. 2013. Sécurisation des entrepots de données: de la conception a l'exploitation. *Rapport de these*.

TRUJILLO, J., SOLER, E., BLANCO, C., AND FERNANDEZ-MEDINA, E. 2009. Designing secure data warehouse by using mda and qvt. *Journal of Universal Computer Science 8 15*, 1607–1641.

Mrs. Amina El ouazzani is a PhD student from National School of Applied Sciences, Tangier Morocco. His research interests are in cloud computing and the confidentiality confidentiality of data warehouses.

Mr. Salah Ouederrou is a PhD student under the supervision of Prof. Hassan BADIR. His research is centred on security and privacy in Internet of Things domain.

Dr. Nouria Harbi is an associate professor at the University Lumiere 2 Lyon, France. she received her PhD in computer science from the INSA Lyon, France in 1990. His research interests are in Multidimensional modeling of complex data, Security of decision-making information systems: Data warehouse access control based on profile management, Data integrity in decision-making information systems, Data mining and security (intrusion detection), Confidentiality and availability of data in Cloud Computing.

Dr. Hassan Badir is an associate professor at the University Abdelmalek Essaâdi , Tetouan Morocco , he received Ph.D. degree in Database complex Design from Claude Bernard University Lyon1 in 2004. He studies in the fields of Computer Science, specifically Business Intelligence, Database, optimization, Cloud Computing and Big data. - Head of Computer Science and Complex Systems Master - Co-Investigator of Human Heredity and Health African Bioinformatics Network - Head and founding member of the Moroccan Innovation and New trends of Informatio System Society.