LA DETECTION DES INFERENCES PAR LA COMBINAISON DE PLUSIEURS PROFILS

Amine El Ouazzani, Nouria Harbi, Hassan Badir

LA DETECTION DES INFERENCES PAR LA COMBINAISON DE PLUSIEURS PROFILS

Amine EL OUAZZANI*, Nouria HARBI**
Hassan BADIR*

* Université Abdelmalek Essaadi ENSA LabTIC 1818 Tanger MAROC
{ a.elouazzani2000 , h.badir } @gmail.com,
http://www.ensat.ac.ma/
**Université Lumière Lyon 2 Laboratoire ERIC 69635 Lyon, Cedex FRANCE
nouria.harbi@univ-lyon2.fr

Résumé. Un Entrepôt de données (ED) regroupe les données sensibles de l'entreprise et les données secrètes sur la vie privée des individus. Ce qui rend la gestion des accès à cette source une tâche difficile qui doit prendre en compte la détection des inférences possibles. Dans ce sens plusieurs auteurs ont proposé des méthodes pour faciliter la gestion des inférences, en analysant les permissions accordées à un utilisateur. Cependant aucun travail n'a traité la gestion des inférences dans le cas d'un utilisateur qui combine entre deux ou plusieurs profils au sein de l'entreprise. Dans cet article, nous allons présenter notre approche qui permet de détecter les déductions possibles entre deux ou plusieurs rôles affectés à un seul utilisateur.

Mots-clés : *Entrepôt de données, Inférence, Données sensibles, Profils utilisateur.*

1 Introduction

Dans cet article, nous nous attachons au champ des contenus implicites dont l'ancrage est invisible en tenant compte des données accessibles. Nous préférerons utiliser le terme d'inférence à celui d'implicite. Nous appellerons inférence toute proposition implicite permettant d'extraire ou déduire une information interdite en combinant des informations accessibles. La détection des inférences par la combinaison des données n'a pas pris beaucoup d'intérêt par les chercheurs malgré son importance pour les EDs qui utilisent plusieurs sources de données. D'après le travail de (Blanco, et al., 2010), les requêtes qui demandent la combinaison de plusieurs éléments multidimensionnels tendent à être plus sensibles qu'à des données séparément. Ces combinaisons qui doivent être modélisées afin d'éviter l'inférence des données non autorisées. Pour avoir un système proactif, notre proposition permet de produire des connaissances sur les inférences non observables entre plusieurs profils combinés par le même utilisateur. Sachant que ce dernier peut avoir plusieurs rôles au sein de l'entreprise. Chaque rôle contient des autorisations. Un utilisateur peut inférer des données sensibles par la combinaison des autorisations des rôles accordés.

Dans cet article, nous présentons notre approche qui se focalise sur la détection des inférences de données sensibles de l'ED par un utilisateur occupant deux ou plusieurs profils au sein de l'entreprise, en utilisant la présentation graphique des permissions de chaque profil et en exploitant les associations des bases de données sources.

2 Etat de l'art et synthèse

2.1 Les travaux existants

- **(Triki, et al., 2013)** ont proposé un modèle pour sécuriser les données multidimensionnelles contre les inférences dans la phase conceptuelle, cette approche suppose que le schéma de DW est déjà conçu. Il permet de détecter les deux types d'inférences :
  - Inférence précise : ou les valeurs des données déduites sont exactes.
  - Inférence partielle : ou les valeurs des données sont partiellement divulguées, c'est à dire que l'utilisateur peut déduire une idée sur la valeur des données.

  Cette approche se compose de trois étapes :
  - **Etape 1.** Un expert de domaine identifie les éléments sensibles à protéger en interrogeant le concepteur de l'ED.
  - **Etape 2.** Construction du graphe des inférences à partir du diagramme de classe, en précisant les éléments qui présentent les inférences précises ou partielles.
  - **Etape 3.** Présentation l'ED avec les annotations UML en mettant en évidence les deux types d'inférences.
- **(Blanco, et al., 2010)** ont Proposé une approche basée sur le diagramme états-transactions pour détecter les inférences au niveau de la conception. Cette proposition se focalise sur les requêtes sensibles et ses évolutions, mais les auteurs ne tiennent pas en compte l'inférence des données à partir des données accessibles. L'approche est présentée sous forme d'un modèle de sécurité OLAP de 3 états :
  - **Modèle statique** : présente le profil UML spécifique aux ED de (Fernández-Medina, et al., 2007), en ajoutant un nouveau genre de règle nommée JR « Joint Rules », qui présente les privilèges nécessaires à certaines combinaisons selon une grammaire spécifique.
  - **Modèle dynamique** : ou bien le modèle états-transactions qui a l'objectif d'enrichir le modèle statique, afin d'assurer que la confidentialité n'est pas compromise en traitant les évolutions des combinaisons définis avec JR à travers l'application des opérations OLAP.
  - **Contrôle de session** : cette étape rend plus d'intérêt aux sessions des utilisateurs afin de les analyser en vérifiant chaque événement pour détecter toute possibilité d'inférence.
- **(Sweeney, 2002)** Décrit un cas réel d'inférence des données, par une démonstration d'identification des données sensibles en se basant sur le croisement des données d'un groupe d'assurance, ces données supposées qu'elles sont anonymes, et une liste d'inscription des électeurs, ce qu'il a permet de détecter le nom de l'ancien gouverneur « William Weld » et ses dossiers médicaux, en reliant les attributs par-

tagés. Il a défini un modèle de protection nommé k-anonymity qui comprend des politiques d'accompagnement, permettant d'éviter des inférences qui sont :

- Trier les lignes des tables à publier d'une façon aléatoire pour ne pas divulguer des informations sensibles.
- Eviter d'avoir une ligne avec une valeur unique dans la table à publier.
- Prendre en compte l'ancienne version des données lors de la réalisation de la nouvelle.

- **(Chen, et al., 2008)** Les auteurs ont proposé un modèle de détection des inférences qui se compose de trois parties :
  - Acquisition des connaissances : basé sur les schémas des bases de données pour extraire les dépendances entre les attributs au sein de la même entité et même entre les entités, les règles et les contraintes pour construire un domaine des connaissances sémantique.
  - Modèle d'inférence sémantique SIM : représente toutes les relations possibles entre les attributs des bases de données. Pour une requête donnée, un graphe des inférences sémantique fournit les canaux d'inférence pour inférer les données sensibles.
  - Détection des violations de sécurité : il se base sur le graphe d'inférence pour comparer la nouvelle requête demandé avec la base des connaissances acquises.
- **(Accorsi, et al., 2013)** Ont proposé une approche dans laquelle les règles d'inférences sont connues par le moteur d'inférence (ces règles peuvent être offertes par un service tiers ou par le consommateur des données). Le diagramme proposé montre un processus de détection des inférences qui se compose de :
  - **Politique composée** : dans lequel l'utilisateur compose la politique et les règles de confidentialité PD={r1,r2,r3…rn}avec ri=(O,L) avec (PD : « Privacy policy of Data » , ri : « Rule » est la règle qui définit pour chaque objet O son niveau de sensibilité L). L'utilisateur peut toujours adapter cette politique en ajoutant d'autres règles. Alors la fermeture d'inférence C par rapport à un domaine D et une politique de confidentialité PD, est l'ensemble de tous les éléments de données inférable à partir du noyau CoreL(PD).
  - **Calcul de la fermeture d'inférence** : Le moteur d'inférence prend cette politique qui calcule à son tour toutes les fermetures d'inférence possibles de la politique entrée en se basant sur un algorithme qui représente les étapes nécessaires.
  - **Le noyau** teste pour chaque élément non noyau s'il est obtenu à partir d'un élément noyau en utilisant les règles de la politique de confidentialité. Le noyau examine les règles de la politique de confidentialité, si ri.L< L, alors les éléments de cette règle appartiendront au noyau.
  - **Visualisation des menaces d'inférence** : Pour distinguer la gravité des menaces, les auteurs ont utilisé le niveau de sensibilité associé à chaque élément inférable.

2.2 Synthèse

La protection des ED contre les accès illégaux s'est fait sentir d'une manière incontestable depuis plusieurs années (Fernandez-Medina, et al., 2006) , (Soler, et al., 2008), (Trujillo, et al., 2009), (Arora, et al., 2016). D'après les auteurs (Eavis, et al., 2012), la modélisation

du contrôle d'accès à l'ED est le processus de construction d'un modèle abstrait qui doit être stockée dans l'ED. Ce modèle est une représentation de la réalité ou d'une partie de la réalité. Suite à l'étude des travaux existants, nous avons constaté les points suivants :

- Bien que les autorisations présentent l'axe principal pour garantir la confidentialité de l'accès à l'ED, cependant l'absence d'une norme qui gère la précision de ces autorisations peut provoquer des incohérences et des inférences comme conséquences. Dans ce sens, on trouve le travail de (Saltor, et al., 2002) qui ont proposé l'utilisation du schéma des autorisations défini pour les bases de données fédérées sans aucune modification pour construire un ED sécurisé, et (Rosenthal, 2000) qui proposent la réécriture des requêtes afin de vérifier que ces dernières respectent les restrictions définit au niveau des sources.
- A noter également, que la notion d'inférence a été citée dans plusieurs travaux en tant qu'élément essentiel pour garantir la confidentialité, et dont la maîtrise est cruciale. Dans ce sens, on trouve le travail de (Triki, et al., 2013) qui ont proposé une approche qui permet de détecter les inférences partielles et précises, (Blanco, et al., 2010) qui proposent une approche basée sur le diagramme d'état-transactions permettant de détecter les inférences dans la phase conceptuelle sans tenir compte la possibilité des inférences à partir des données accessibles. Néanmoins, malgré les risques élevés d'inférences, il n'est pas suffisamment pris en compte dans la phase conceptuelle.

3 La détection des inférences par la combinaison de plusieurs profils

3.1 Concept de base

Afin de permettre une extraction automatique des inférences à partir des permissions autorisées, nous proposons des règles à vérifier en se basant sur la présentation graphiques des profils accordés à un utilisateur et les liens entre les données en utilisant le diagramme de classe source. Cette approche est la suite du travail de (Triki, et al., 2013) qui propose une méthode de détection des inférences précises et partielles, cependant notre proposition consiste à détecter les combinaisons sensibles.

Sachant qu'un utilisateur peut avoir un ou plusieurs rôles au sein de l'entreprise, ce dernier accède à l'ED avec un ou plusieurs profils. Le but de notre système de détection des inférences est de détecter si l'utilisateur qui a plus qu'un profil, peut déduire indirectement des informations non autorisées en utilisant deux ou plusieurs requêtes depuis deux ou plusieurs profils différents

3.2 Description de l'architecture proposée

L'architecture globale proposée (figure 1) présente notre modèle de contrôle d'accès à base des profils utilisateur qui se compose de trois modules. Le module 1 s'occupe de la classification dynamique des niveaux de sensibilité des données de l'entrepôt, en se basant sur les profils utilisateurs (El Ouazzani, et al., 2016). Le module 2 de notre architecture présente le sujet de cet article. Il se focalise sur la détection des inférences par la combinaison de plusieurs profils par un même utilisateur. Afin d'envoyer les combinaisons sensibles détectées au propriétaire de données.

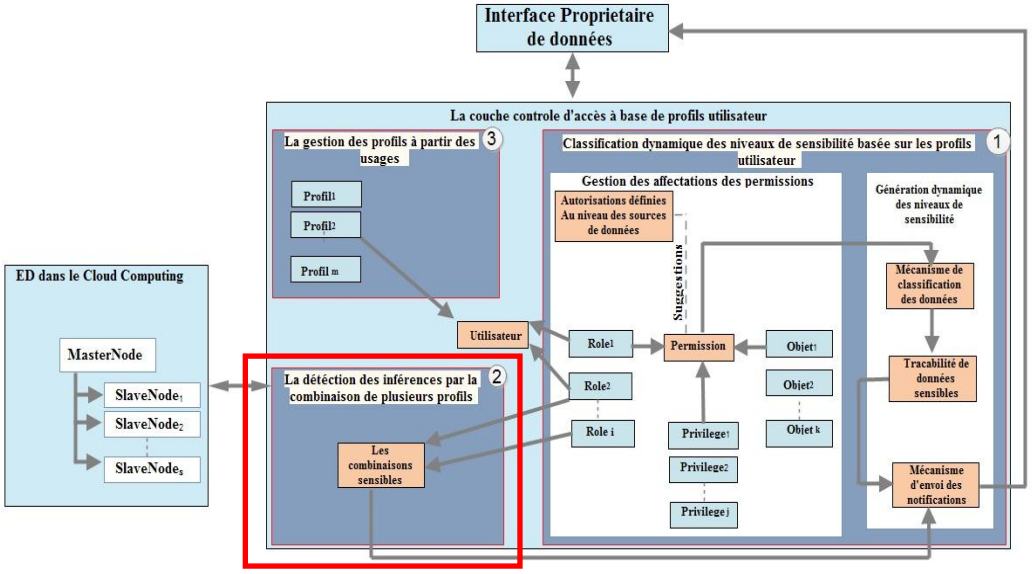

Figure 1 Architecture globale proposée

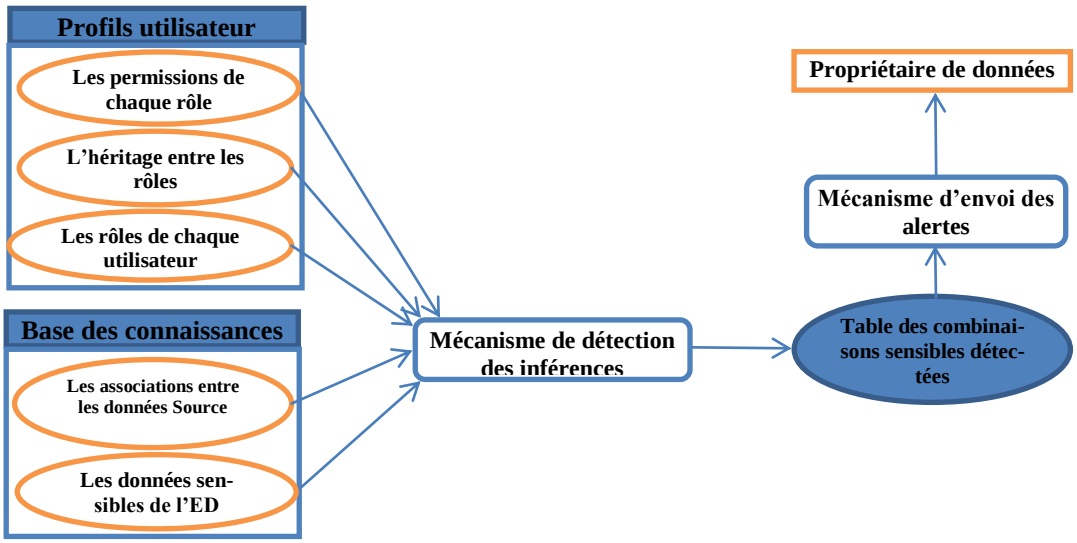

Figure 2 la détection des inférences par la combinaison de plusieurs profils

La **figure 2** présente l'architecture détaillée de notre module de détection des inférences, qui permet d'analyser les permissions de chaque profil, afin de détecter les combinaisons sensibles, il se base sur deux entrées qui sont :

- **Les profils utilisateur :** présentent les utilisateurs qui endossent des rôles qui leurs sont attribués par le propriétaire de données. Ces rôles sont organisés en hiérarchie, et ils disposent des permissions qui ne peuvent pas être accordées directement aux utilisateurs.
- **La base des connaissances :** ce sont les données sensibles de l'ED à protéger contre les inférences, et les associations entre les données selon le diagramme de classe de la base de données source.

3.3 Mécanisme de détection des inférences

L'analyse des permissions accordées à un utilisateur selon ses rôles est un processus important pour le propriétaire de données et pour l'entreprise. Elle sert à limiter les risques et améliorer la qualité de l'affectation des permissions. Elle permet d'identifier les anomalies dans le processus de l'affectation des permissions afin de les corriger. Elle peut être considérée comme un moyen de prévention pour le propriétaire de données en détectant les différents risques d'inférence à partir des permissions accordées en faisant intervenir le diagramme de classe des sources de données.

3.3.1 Utilisation des graphes pour la présentation des profils utilisateurs

Afin de modéliser les profils de chaque utilisateur, nous nous appuyons sur la notion des graphes que nous trouvons dans la littérature (Triki, et al., 2013) , (De Capitani di Vimercati, et al., 2008). Une permission p est une règle de la forme [O,R]→S qui stipule qu'un utilisateur S qui occupe le rôle R, a le droit d'accéder à l'objet O.

Dans notre modèle les profils utilisateur sont présentés par des graphes où les nœuds sont les éléments d'un profil (utilisateur, rôle, objets). Un objet de l'ED peut être une colonne, une table ou une valeur d'une colonne. Un arc (Tableau 1) entre un rôle et les objets représente les droits d'accès d'un rôle sur l'ensemble des objets de l'ED. Un arc entre deux objets représente une association selon le diagramme de classe de la base de données source. Par contre un arc entre deux rôles représente un héritage. Les nœuds du premier niveau du graphe présentent les utilisateurs, les nœuds du deuxième niveau du graphe présentent les rôles d'un utilisateur et les autres niveaux contiennent les objets de l'ED. Les nœuds coloriés sont les objets de l'ED sensibles.

Elément	*Signification*
S	Un nœud Représentant un utilisateur
R	Un nœud Représentant un rôle
O	Un nœud Représentant un objet de l'ED non sensible
O	Un nœud Représentant un objet de l'ED sensible
→	Un arc en ligne continu entre un rôle et des objets de l'ED présente les permissions d'un rôle. Un arc en ligne continu entre deux objets de l'ED indique une association source entre deux objets autorisés à l'utilisateur
- - - →	Un arc en ligne continu entre deux rôles indique un héritage entre deux rôles. Un arc pointillé indiquant une association qui provoque une inférence depuis un objet autorisée vers un objet non autorisée à l'utilisateur

TAB 1 *Signification des éléments graphiques*

Exemple : Prenons un exemple simple pour illustrer la construction d'un graphe. Soit l'utilisateur Alice qui occupe deux rôles au sein de l'entreprise {Medical Secretariat Analyzer, Manager Analyzer}.

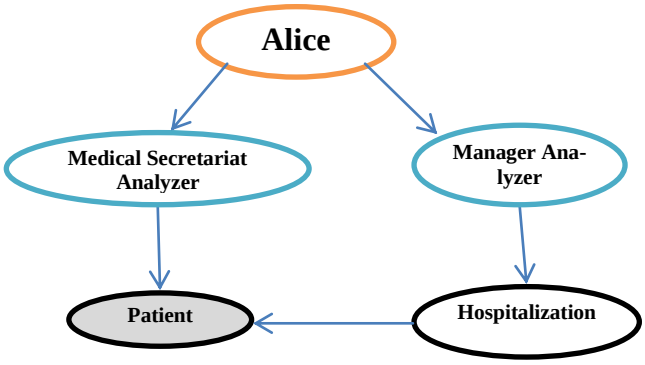

Figure 3 Exemple d'un graphe

La permission P1 : [Patient, Medical Secretariat Analyzer] →Alice signifie que l'utilisateur Alice qui occupe le rôle « Medical Secretariat Analyzer » a le droit de consulter l'objet sensible « Patient ».

La permission P2 : [Hospitalization, Manager Analyzer] →Alice signifie que l'utilisateur Alice qui occupe le rôle d'un « Manager Analyzer » a le droit de consulter l'objet non sensible « Hospitalizaion ».

La figure 3 schématise les permissions de chaque rôle occupé par l'utilisateur Alice. Seul l'objet « Patient » du graphe est colorié, car il présente un objet sensible. L'arc entre les

objets « Hospitalization » et « Patient » présente un lien entre eux selon le diagramme de classe des sources, indiquant que l'objet « Hospitalization » contient la clé primaire de l'objet « Patient » qui est le CIN (Code d'Identité Nationale), et par conséquence c'est une inférence.

3.3.2 Définition des données sensibles

En général, les données sensibles sont les données classées par le premier module de notre architecture globale (figure 1), dont leur niveau de sensibilité est supérieur au seuil fixé par le propriétaire de données (El Ouazzani, et al., 2016). Les données à protéger contre les inférences sont les données sensibles non autorisées à un utilisateur qui peuvent être déduites en combinant des permissions des profils accordés à ce dernier.

3.3.3 Règles d'inférence entre les profils utilisateurs

Afin de vérifier la cohérence des permissions accordées à un utilisateur selon plusieurs profils, sans risques d'inférer des données sensibles non autorisées, nous avons défini cinq règles permettant de définir les combinaisons sensibles.

1. **Par un intermédiaire non autorisé**

C'est une règle qui permet de détecter une inférence par la combinaison des données permises (Figure 4), en utilisant un intermédiaire non autorisé O3 qui peut être sensible ou non sensible contenant un champ commun être deux permissions autorisées.

Dans ce cas la combinaison (O1, O2) est une combinaison sensible. Et par conséquence il va falloir vérifier l'existence d'un lien entre ces objets au niveau de l'ED.

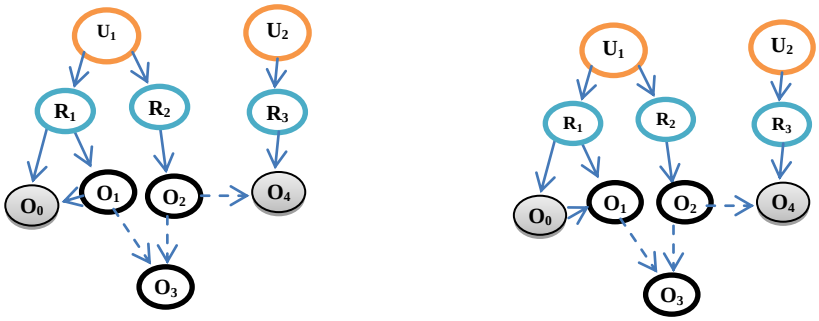

Figure 4 Règle d'inférence par un intermédiaire non autorisé

2. **Par un passage direct**

C'est une règle qui permet de détecter une inférence par la combinaison de plusieurs autorisations d'accès. Selon la figure 5, et afin de déduire une information sur l'objet sensible non autorisé O3, l'utilisateur U1 peut utiliser une requête sur l'objet O2 qui présente un passage entre les objets sensibles permis à l'utilisateur selon le rôle R1 et l'objet sensible O3.

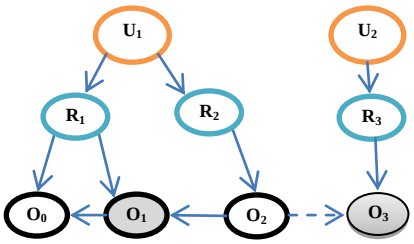

Figure 5 Règle d'inférence par un passage Direct

Dans ce cas et selon cette présentation graphique, la combinaison sensible se compose des objets (O1, O2). Et par conséquence il va falloir vérifier l'existence d'un lien entre ces objets au niveau de l'ED

3. Par un intermédiaire autorisé

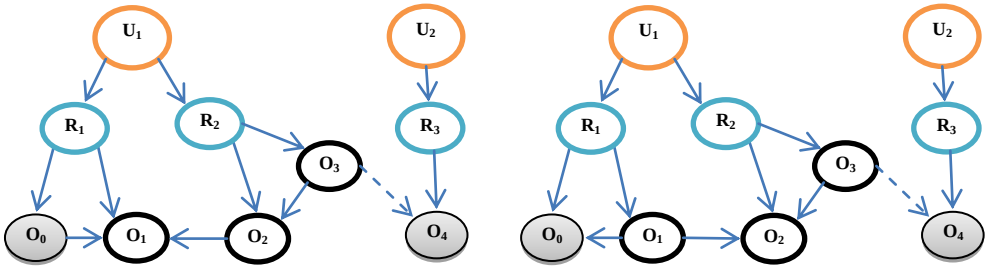

Figure 6 Règle d'inférence par un intermédiaire autorisé

C'est une règle qui permet de détecter une inférence en utilisant un champ commun entre deux permissions affecter à un utilisateur selon un ou plusieurs profils, en utilisant des requêtes avec le champ commun. D'après la figure 6 la combinaison sensible se compose de l'objet O1 et l'objet intermédiaire O3. Et par conséquence il va falloir vérifier l'existence d'un lien entre ces objets au niveau de l'ED.

4. Par un passage invisible

La règle d'inférence par un passage invisible selon la présentation graphique (Figure 7), permet de déterminer une combinaison sensible sans aucun intermédiaire visible. En ouvrant une session avec le rôle R2, l'utilisateur U1 peut exécuter une requête avec un critère sur le champ commun entre O2 et O3. Ensuite, l'utilisateur a la possibilité d'ouvrir une session avec le rôle R1 afin d'exécuter une autre requête sur l'objet O1, avec un critère contenant le résultat de la première requête. Ce qui permet de créer une inférence de l'objet O3 qui n'est pas autorisé.

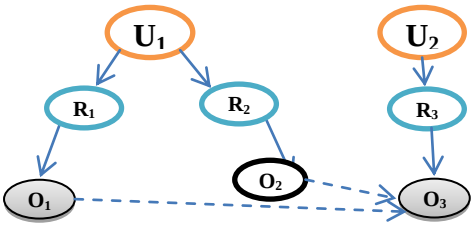

Figure 7 Règle d'inférence par un passage invisible

Donc, nous pouvons déduire que la combinaison (O1 , O2) est une combinaison sensible. Et par conséquence il va falloir vérifier l'existence d'un lien entre ces objets au niveau de l'ED.

5. Par héritage

Cette règle permet de détecter une inférence par héritage. Selon la figure 8, le rôle R3 a le

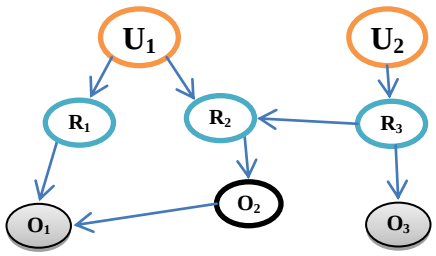

Figure 8 Règle d'inférence par héritage

droit de lire l'objet O3, et il hérite les permissions du rôle R2. Ce dernier a la permission d'accéder à l'objet O2, et d'après le diagramme de classe source, l'objet O2 et en association avec l'objet O1. Et par conséquence l'utilisateur U2 peut exécuter des requêtes avec le rôle R3, afin d'utiliser le résultat dans une autre requête avec le R2 pour déduire une information sur l'objet sensible non autorisé O1. Nous pouvons constater que la combinaison (O2, O3) est une combinaison sensible. Et par conséquence il va falloir vérifier l'existence d'un lien entre ces objets au niveau de l'ED.

3.3.4 Vérification des règles

La détection d'une inférence selon les règles proposées, en utilisant le diagramme de classe sources, implique la vérification d'un lien entre objets de la règle selon le schéma de l'ED afin de détecter des inférences non autorisées.

4 Conclusion et perspectives

Dans cet article, nous avons défini la problématique de la détection des inférences par la combinaison des permissions autorisées. Cette problématique n'a pas pris un intérêt par les chercheurs malgré son importance de bien vérifier la cohérence des permissions selon un ou plusieurs profils affectés à un seul utilisateur. Dans ce sens, nous avons proposé des règles permettant de détecter les permissions sensibles qui peuvent inférer des données sensibles non autorisées. Ce qui peut aider le propriétaire de données à bien contrôler les permissions accordées. Avec notre exemple d'interprétation, nous avons présenté les règles de détection des inférences proposées dans un cas réel afin de bien les expliquer. Parmi nos perspectives, nous avons l'intention de chercher et améliorer d'autres règles, permettant de détecter des inférences entre plusieurs profils d'un seul utilisateur de l'ED.

Références

Accorsi, R. et Müller, G. 2013. Preventive inference control in data-centric business models. s.l. : Security and Privacy Workshops (SPW), 2013 IEEE (pp. 28-33). IEEE. , 2013.

Arora, D. et Kumar, U. 2016. Protecting Sensitive Warehouse Data through UML based Modeling. s.l. : Proceedings of the International Conference on Informatics and Analytics (p. 31). ACM., 2016.

Blanco, C., et al. 2010. Towards the secure modelling of olap users behaviour. 2010.

Chen, Y et Chu, W. W. 2008. Protection of database security via collaborative inference detection. IEEE Transactions on Knowledge and Data Engineering, 20(8), 1013-1027. : s.n., 2008.

De Capitani di Vimercati, S., et al. 2008. Assessing query privileges via safe and efficient permission composition. . s.l. : Proceedings of the 15th ACM conference on Computer and communications secur, 2008.

Eavis, T. et Althamimi, A. 2012. Olap authentication and authorization via queryre-writing. s.l. : The Fourth International Conference on Advances in Databases, Knowledge, and Data Applications, 130–139, 2012.

El Ouazzani, A., et al. 2016 . Dynamic management of data warehouse security levels based on user profiles. s.l. : Information Science and Technology (CiSt). 4th IEEE International Colloquium on (pp. 59-64). IEEE., 2016 .

Fernandez-Medina, E., et al. 2006. Access control and audit model for the multidimensional modeling of dws. . s.l. : Decision Support Systems, 1270–1289., 2006.

Fernández-Medina, E., et al. 2007. Developing secure data warehouses with a UML extension. s.l. : Information Systems, 32(6), 826-856., 2007.

Rosenthal, A. et S. Sciore. 2000. . View security as the basis for data warehouse security. . s.l. : In DMDW (p. 8)., 2000.

Soler, E., Stefanov, V. et Mazon, N.J. 2008. Towards Comprehensive Requirement Analysis for Data Warehouses: Considering Security Requirements, pp. 104–111. IEEE,. Los Alamitos : s.n., 2008.

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

Triki, S., et al. 2013. Sécurisation des entrepôts de données: de la conception à l'exploitation. Lumiere II Lyon : Rapport de thèse., 2013.

Trujillo, J., et al. 2009. A UML 2.0 profile to define security requirements for Data Warehouses. . s.l. : Computer Standards & Interfaces, 31(5), 969-983., 2009.

La detection des inférences par la combinaison de plusieurs profils

Summary

A Data Warehouse is a collection of sensitive and secret company data on the privacy of individuals. This makes the management of access to this source a difficult task that must take into account the detection of possible inferences. In this sense several authors have proposed methods to facilitate the management of inferences, by analyzing the permissions granted to a user. However, no work has dealt with inference management in the case of a user who combines two or more profiles within the company. In this article, we will present our approach to detect possible inferences between two or more roles assigned to a single user.