Plus précisément, dans les deux points rouges illustrés dans la Figure 6, une fonction de matching est exécutée afin d'établir en fonction du profil utilisateur et de sa situation s'il a le droit d'accéder le contenu :

F : (Up; Situation; Policy) => Permit/Deny;

ou Up est le profil utilisateur, la Situation est un niveau d'urgence (explicitement fournit par l'utilisateur ou inféré à partir de la localisation et du contexte) et Policy est un ensemble de règles qui définit la politique de sécurité du système.
Le profil utilisateur est définit de la façon suivante:

Up =< Uid; Name; Login; Password; RoleId >

ou Uid est le id de l'utilisateur, Name est le nom de l'utilisateur, Login est son login, Password est le mot de passe de l'utilisateur et RoleId représente le rôle qui est associé à l'utilisateur.
Une règle d'accès est définie de la façon suivante:

Rule =< RuleId;RoleId; Action;Context; Permission >

Une règle définit une certaine permission (Permit ou Deny) pour un certain rôle (i.e, RoleID) et Action (e.g. indexation explicite, visualisation de l'objet) dans un certain contexte (e.g., localisation).

Dans la suite, nous introduisons la fonctionnalité détaillée de PSQRS.

4.3. L'architecture PSQRS

Comme présenté en Figure 7, l'architecture que nous proposons vise à étendre le modèle de prise de décision dans XACML en ajoutant une couche adaptative réalisée par un mécanisme de réécriture de la requête de l'utilisateur dans le cas où elle est refusée par le PDP (Al Kukhun et Sèdes, 2008).

Le système récupère les contraintes contextuelles de l'utilisateur et les reçoit avec une étape d'authentification (1). Puis, quand l'utilisateur lance sa requête, le système la communiquera au générateur de requête (2) qui la traduira vers une requête R de format XACML. Celui-ci prend en compte cette demande et la combine avec les contraintes contextuelles puis l'envoie vers l'Evaluateur de requêtes (3) qui joue le rôle d'un PDP et suit le processus normal de XACML.

Selon les droits d'accès de l'utilisateur (précisés par les politiques d'accès sauvegardées dans les PAPs – voir le schéma XACML en section 3.3), le système répond à cette demande soit en permettant l'utilisateur d'accéder à la ressource demandée (4a), ou en lui répondant avec un refus d'accès (4b). C'est dans ce dernier cas que notre mécanisme adaptatif intervient pour étudier la situation dans laquelle l'utilisateur a consulté le système. Cette situation est définie par le Fournisseur de Sensibilité de la Situation (5 et 6) qui autorise la régénération de la requête R' dans le cas d'une situation d'urgence par exemple.

Cette régénération ou réécriture de requêtes est réalisée grâce au Fournisseur de similarité (7 et 8) qui prend les contraintes contextuelles de l'utilisateur comme un point de départ pour la recherche des documents ou des services autorisés et qui ont des similarités de contenu ou de fonctionnement avec la ressource initialement demandée par l'utilisateur et qui été jugé comme non autorisée.

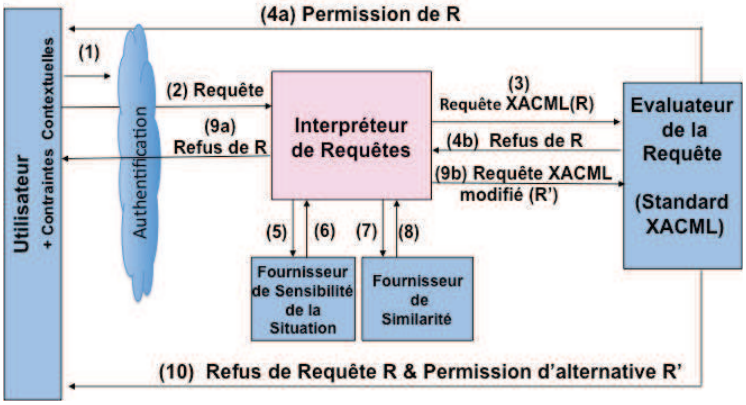

Figure 7. L'architecture de PSQRS

Cette étape est destinée à restituer à l'interpréteur de requêtes des ressources alternatives similaires. Dans le cas où le système n'aura pas des propositions, l'interpréteur des requêtes va envoyer à l'utilisateur un refus d'accès (9a) – cas classique. En revanche, dans le cas où le système trouve des ressources similaires, le Générateur de la Requête réécrit la requête initiale en remplaçant la ressource demandée par la nouvelle ressource jugée similaire (par notre Fournisseur de Similarité) puis l'envoie vers l'Evaluateur de Requêtes qui réévalue la nouvelle requête R' (9b) et répond l'utilisateur avec un refus d'accès pour sa demande initiale et une permission pour accéder aux ressources alternatives (10).

Dans la suite nous présentons plus en détail la façon dont nous avons intégré l'approche PSQRS dans le système LINDO.

4.4. La fusion PSQRS et du système LINDO

Afin d'inclure la couche de contrôle d'accès dans l'architecture LINDO, plusieurs changement doivent être faites à l'intérieur du système. Plus précisément, d'un point de vue fonctionnel, chaque fois qu'une demande d'accès est faite (les points rouges de la Figure 6), le système PSQRS est utilisé avant d'exécuter l'indexation explicite et avant d'afficher les résultats. D'un point de vue architecturel, l'architecture du système présentée en détail dans (Brut et al. 2011) a été étendue avec plusieurs modules et fonctionnalités.

En conséquence, dans la Figure 8, les modules qui ont été ajoutés ou modifiés sont affichés en rouge. La plupart des changements se passent au niveau du serveur central :

- Le module d'authentification (Authentification Module) a été ajouté pour vérifier les utilisateurs qui essayent d'interroger le système. Ce module communique avec le module Metadata Engine afin de déterminer si l'utilisateur est «reconnu » par le système et de retrouver son profil et son contexte ;

- Le module Access Control a été ajouté. Ce module contient les sous modules PSQRS Query Interpreter (Interpréteur de requêtes), Sensitivity Analyser (Fournisseur de Sensibilité de la Situation) et Similarity Provider (Fournisseur de la similarité). Toute information qui est envoyée à l'utilisateur passe par ce module afin d'appliquer une éventuelle adaptation du contrôle d'accès ;

- Le module Terminal Interface a été modifié afin de capturer le contexte et la situation de l'utilisateur ;

- une base de données avec les rôles RBAC, les règles et les profils des utilisateurs a été incluse dans le module Metadata Engine. Cette base de données contient aussi les règles d'adaptation d'accès ;

- Le module Query Analyser a été integré dans le Request Processor ;

- Dans la structure du Remote Server, seulement le module Access Manager a été modifié. En effet, une nouvelle fonctionnalité a été ajouté à ce module : l'exécution des algorithmes d'indexation ;

Pour chaque rôle RBAC identifié, les droits d'accès aux contenus multimédias, et à l'exécution des certains algorithmes d'indexation sont spécifiés en concordance avec le contexte de l'utilisateur au moment ou il interroge le système. La situation de l'utilisateur est capturée de façon implicite (en analysant le contexte) ou de façon explicite (spécifiée par l'utilisateur). Afin de fournir des alternatives à l'utilisateur en fonction de sa situation, le module Similarity Provider peut sélectionner d'autres contenus ou exécuter des algorithmes qui extraient des informations du contenu ou modifient le contenu afin de respecter la politique de sécurité du système.

5. Illustration dans un système de vidéosurveillance

Dans cette section, nous présentons un exemple où notre proposition est utilisée pour pallier le manque des réponses restituées par le système. Comme nous allons l'illustrer, le système va modifier le traitement de la requête et l'adaptation de prise de décision d'accès selon le niveau d'importance de la situation.

Ce cas d'utilisation concerne une compagnie de transport en commun qui a installé des cameras de vidéo surveillance dans les bus et rames de métro, dans les stations et à coté des machines aux billets. Le système qui gère ces cameras peut être utilisé par les agents de sécurité et le policier. En conséquence on peut identifier deux rôles : agent de sécurité et policier. L'accès au système peut être fait depuis la chambre de contrôle, ou depuis les stations en utilisant un dispositif mobile. Pour chaque rôle un ensemble de restrictions peuvent être spécifiées.

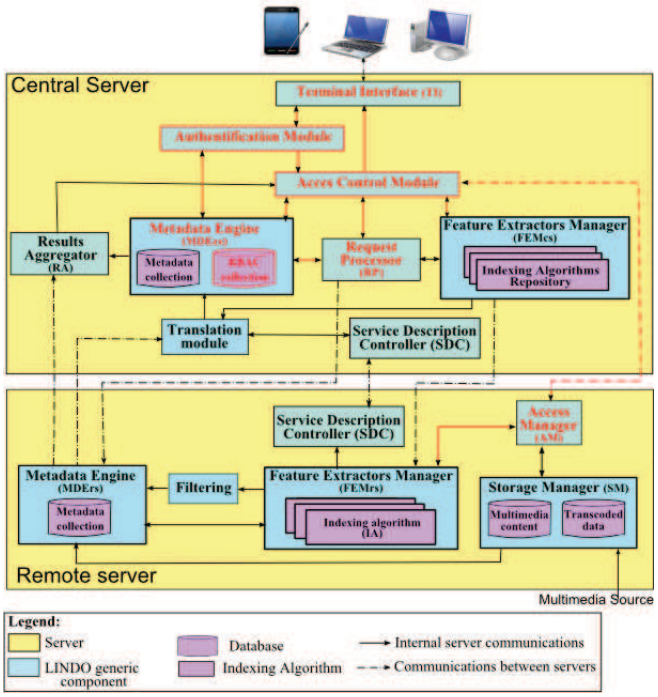

Figure 8: L'architecture LINDO avec l'approche PSQRS incorporée

Tableau 1: Exemples de droits d'accès

Rôle	Contexte de l'utilisateur	Contenus	Action	
			Voir les visages	Indexation explicite
Agent de sécurité	Chambre de contrôle	Tous	Permettre	Permettre
	Station	Cameras métro	Refuser	Permettre, seulement Object Tracking
		Cameras bus	Refuser	Refuser
Policier	Chambre de contrôle	Tous	Permettre	Permettre
	Station	Cameras métro	Permettre	Permettre, Object et Person Tracking
		Cameras bus	Permettre	Refuser

Le Tableau 1 fournit quelques exemples de droits d'accès donnés à chaque rôle, dans un certain contexte pour réaliser une certaine action sous des contenus multimédias.

Scénario : En prenant le métro de la station Trocadéro vers la place d'Italie à 14h15, Hélène a oublié son sac rouge sur un banc d'attente sur un quai. Dès qu'elle s'en est rendue compte, elle est sortie et s'est rendue au guichet de la station pour signaler le problème.

Le traitement typique d'une telle situation passe par l'agent de service clientèle qui ouvre un dossier, prend les descriptifs de l'objet perdu et les transmet à l'agent de sécurité sur place. Ce dernier va suivre différentes étapes pour retrouver l'objet : il va vérifier si l'objet a été déjà retrouvé ou remis au service par quelqu'un. Sinon, il va essayer de consulter le système de vidéo surveillance pour vérifier si l'objet est toujours au même endroit.

5.1. Le traitement typique d'une requête selon LINDO

La Figure 9 montre l'interprétation typique réalisée par le système de recherche d'information fourni par le système LINDO. La requête lancée sera traitée et parcourue afin d'extraire les mots-clés qui ensuite seront reformulés sous forme d'une requête XML.

Après l'extraction des mots-clés de la requête, le traitement de la requête va procéder à la localisation des serveurs gérant les différents flux capturés par les caméras situées dans les quais d'attente de la station Trocadéro. Puis, une étape de filtrage sera effectuée pour restreindre la recherche dans les parties acquises entre 14h00 et 15h00. Le système va déterminer, ensuite, une liste d'algorithmes d'indexation appropriée à l'ensemble des besoins, des propriétés et des contextes exprimés dans la requête. Cette étape va générer les métadonnées liées à la requête.

Query: Trouve toutes les vidéos qui contiennent un sac rouge, oublié à l'arrêt de métro Trocadéro, Paris jeudi 2 février, entre 2:00pm et maintenant (3:00pm).

Dans ce scénario, les informations demandées sont basiques, la requête sera traitée à l'aide des résultats d'indexation réalisés par les algorithmes implicites placés au niveau du serveur central. Le système va poursuivre la recherche pour trouver un objet rouge dans les métadonnées décrivant les segments choisis.

Un processus de filtrage additionnel est appliqué pour la prise en compte des règles de contrôle d'accès. En examinant les droits d'accès de l'agent de sécurité, nous trouvons qu'il n'a pas l'autorisation de consulter des vidéos qui affichent les visages des passagers, ni d'utiliser les algorithmes d'indexation explicites existant au niveau des serveurs distants. Par conséquent, le système filtre les ressources en éliminant les parties qui contiennent des visages de personnes et enfin, renvoie à l'utilisateur des segments qui contiennent un objet rouge (s'il en existe).

5.2. Traitement adaptatif sensible au contexte et à la situation en utilisant PSQRS

L'analyse des résultats restitués à l'agent de sécurité dans ce cas, montre que ces derniers sont insuffisants. Notre proposition intervient à ce niveau afin d'améliorer

la qualité de service et d'offrir à l'utilisateur plus de ressources accessibles sans dépasser les droits d'accès imposés sur la consultation des ressources de données.

```
<UserQuery>
    <QueryInText> Trouve toutes les vidéos qui contiennent un sac rouge, oublié à l'arrêt de métro Trocadéro, Paris jeudi 2 février, entre 2:00pm et maintenant (3:00pm).</QueryInText>
    <MediaLocation> métro Trocadéro, Paris </MediaLocation>
    <MediaFormat>vidéo</MediaFormat>
    <TimeSpan>
        <From>2012-02-02T14:00:00</From>
        <To> 2012-02-02T15:00:00</To>
    </TimeSpan>
</UserQuery>
```

Figure 9. Structure XML d'une requête

L'utilisation de l'architecture PSQRS permettra au système de modifier le niveau d'accessibilité et d'adapter les permissions offertes à l'agent de sécurité selon son contexte et l'importance de la situation de consultation.

L'utilisation de cette solution est liée au déclenchement de la reconnaissance d'une situation ou d'un contexte par le système. Dans ce scénario, la situation sera reconnue depuis l'identifiant du dossier « objet perdu ».

L'implémentation adaptative de notre proposition est réalisée par le système PSQRS qui adapte la prise de décision par la réécriture des requêtes XACML. Cette solution a prouvé son efficacité par sa capacité à fournir une prise de décision d'accès à partir des politiques distribuées à prendre en compte des éléments contextuels liés à la requête.

Par conséquent, cette simple requête lancée par l'agent de sécurité (composée par des mots-clés décrivant le contenu recherché, voir Figure 9) sera incluse dans une demande d'accès sous forme d'une requête XACML plus structurée et enrichie des métadonnées (décrivant les contraintes contextuelles de l'utilisateur, son rôle, l'importance de la situation dans laquelle il consulte le système, voir Figure 10).

Le niveau d'importance de la situation va servir à déterminer le niveau d'adaptation qui sera réalisé ensuite. L'activation du mode de recherche adaptatif sera communiquée à partir de la réponse XACML sous la forme d'une « obligation » qui accompagne la réponse, voir Figure 11.

```
<Requestxmlns="urn:oasis:names:tc:xacml:2.0:context:schema:os"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:context:schema:os
http://docs.oasisopen.org/xacml/access_control-xacml-2.0-context-schema-os.xsd">
    <Subject>
        <AttributeAttributeId="urn:oasis:names:tc:xacml:2.0:subject:subject-id"
DataType="http://www.w3.org/2001/XMLSchema#string">
            <AttributeValue>John Smith</AttributeValue>
        </Attribute>
        <AttributeAttributeId="urn:oasis:names:tc:xacml:2.0:subject:role"
DataType="http://www.w3.org/2001/XMLSchema#anyURI">
            <AttributeValue>Agent de sécurité</AttributeValue>
        </Attribute>
        <Attribute
AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:securityAgent-id"
DataType="http://www.w3.org/2001/XMLSchema#string">
            <AttributeValue>sa2023</AttributeValue>
        </Attribute>
    </Subject>
    <Resource>
        <ResourceContent>
            <UserQuery>
                <QueryInText> Trouve toutes les vidéos qui contiennent un sac rouge, oublié
à l'arrêt de métro Trocadéro, Paris jeudi 2 février, entre 2:00pm et maintenant
(3:00pm).</QueryInText>
                <MediaLocation> métro Trocadéro, Paris </MediaLocation>
                <MediaFormat>vidéo</MediaFormat>
                <TimeSpan>
                    <From>2012-02-02T14:00:00</From>
                    <To> 2012-02-02T15:00:00</To>
                </TimeSpan>
            </UserQuery>
        </ResourceContent>
    </Resource>
    <Action>
        <AttributeAttributeId="urn:oasis:names:tc:xacml:2.0:action:action-id"
DataType="http://www.w3.org/2001/XMLSchema#string">
            <AttributeValue>Lire</AttributeValue>
        </Attribute>
    </Action>
    <Environment>
        <Attribute AttributeId="urn:oasis:names:tc:xacml:2.0:environment:environment-id"
DataType="http://www.w3.org/2001/XMLSchema#string">
            <AttributeValue>Situation</AttributeValue>
        </Attribute>
        <Attribute AttributeId="urn:oasis:names:tc:xacml:2.0:environment:situation-id"
DataType="http://www.w3.org/2001/XMLSchema#string">
            <AttributeValue>Objet oublié</AttributeValue>
        </Attribute>
        <Attribute AttributeId="urn:oasis:names:tc:xacml:2.0:environment:sitLevel-id"
DataType="http://www.w3.org/2001/XMLSchema#string">
            <AttributeValue>1</AttributeValue>
        </Attribute>
    </Environment>
</Request>
```

Figure 10. Requête XACML englobant la requête de l'utilisateur, son rôle, son contexte et sa situation

```
<Response>
  <Result>
    <Decision>Deny</Decision>
    <Status>
      <StatusCode Value="urn:oasis:names:tc:xacml:2.0:status:ok"/>
    </Status>
    <Obligation FulfillOn="Deny" ObligationId="ApplyAdaptiveQueryingMode">
      <AttributeAssignment AttributeId="AQM"
            DataType="http://www.w3.org/2001/XMLSchema#string"> On
      </AttributeAssignment>
    </Obligation>
  </Result>
</Response>
```

Figure 11. Réponse XACML avec les obligations à réaliser

Le déclenchement du mode adaptatif (Adaptive Querying Mode) va changer le processus du traitement de la requête afin d'assurer la réussite de la recherche en proposant des solutions adaptatives.

Cette solution est réalisée dans le système PSQRS au niveau du Fournisseur de Sensibilité de la Situation qui détecte la situation puis, s'oriente vers le Fournisseur de Similarité pour réaliser la réécriture de la requête (c.f. Figure 7).

Dans le cas où la situation de consultation est normale (sitLevel-id = 0 dans la requête XACML), le système réalisera une reformulation sémantique des mots-clés de la requête en utilisant des mots similaires ou des concepts plus génériques au niveau de Fournisseur de similarité. Un travail similaire a été introduit dans (Al Kukhun et Sèdes, 2008), l'objectif étant d'augmenter les chances de restitution des résultats aux utilisateurs malgré les challenges de sécurité.

La reformulation sémantique peut être réalisée avec l'aide d'un dictionnaire lexical standard tel que WordNet . Par exemple, le mot « sac » peut être remplacé par différents synonymes {bagage, cabas, sacoche, etc.}. L'emploi de la reformulation a été également proposé par d'autres travaux de l'équipe (Brut et al., 2011a)

Au niveau du traitement du scénario courant, le niveau d'importance de la situation est plus élevé (sitLevel-id = 1 dans la requête XACML présentée dans la Figure 11). De ce fait, le Fournisseur de Similarité sera remplacé par un Fournisseur de Solutions Adaptatives. Ce composant va réaliser une adaptation automatique ou assister l'utilisateur pour adapter sa requête en lui fournissant des propositions de solutions adaptatives sauvegardées dans une base de données prédéfinie. Le tableau 1 montre des exemples de solutions proposées par le système.

L'alimentation de la base de données peut aussi être effectuée par une méthode d'apprentissage automatique à partir des solutions proposées par les utilisateurs en fonction des situations rencontrées en temps réel. La réussite de telles solutions adaptatives ou alternatives (proposées par les utilisateurs) sera plus probable si on connaît la cause d'un refus d'accès. Les messages d'erreur qui accompagnent souvent les réponses négatives retournées peuvent servir d'indicateurs pour trouver des solutions alternatives.

Tableau 2. Exemples de solutions adaptatives proposées par notre système

Problème	Solution Adaptative
Loi d'anonymat imposée au contenu des ressources vidéo capturées	
Visage non-autorisé	Afficher le contenu après l'emploi d'un algorithme qui floute les visages.
Voix non-autorisée	Utiliser un algorithme de transcription textuelle « speech-to-text ».
Volume de vidéo	
Manque de capacité de stockage sur la machine de l'utilisateur.	Utiliser un algorithme de compression ou de conversion vers un format plus léger.
Format non supporté par la machine.	Utiliser un algorithme de conversion vers un format compatible.
Difficulté de téléchargement due à la faiblesse de la bande passante du réseau.	Utiliser un algorithme de synthèse du contenu de vidéo ou héberger les ressources et les consulter à partir d'un espace externe « Cloud computing ».

Par conséquent, la solution adaptative pour cet exemple va modifier le processus du traitement et va : (i) négliger l'étape de filtrage chargée d'imposer les contraintes du contrôle d'accès et (ii) la remplacer par une étape adaptative liée à la présentation de ressources ayant du contenu non-autorisé.

En appliquant ce processus au scénario décrit précédemment, le système va restituer - s'ils existent- les segments vidéo capturés dans la station Trocadéro entre 14h00 et 15h00 et qui contiennent un objet rouge.

Ces résultats seront classifiés de façon à détecter les parties non autorisées (contenant des visages de personnes) et c'est là que le système appliquera un processus de filtrage qui adapte l'affichage pour qu'il soit conforme aux restrictions d'accès imposées par le système.

L'adaptation de présentation consistera à la détection des visages puis à l'utilisation d'un algorithme qui floute les visages apparaissant dans ces segments afin de les présenter à l'utilisateur en respectant les règles d'accès.

6. Conclusion

Le projet LINDO a introduit une architecture de gestion des données multimédias distribuées qui facilite la consommation réduite des ressources en implémentant une technique d'indexation différée et distribuée. Dans cet article, nous avons montrée comment il a été étendu par une couche de contrôle d'accès adaptatif afin d'assurer l'accès aux ressources dans des contextes pervasifs (dans lesquels l'utilisateur pourra accéder aux ressources de données à n'importe quel moment, depuis n'importe où et n'importe comment).

Afin d'atteindre notre objectif, nous avons employé une solution adaptative du contrôle d'accès sensible au contexte et à la situation de consultation. La solution surmonte les refus d'accès survenus suite à des requêtes utilisateurs en modifiant le processus de traitement des requêtes et en proposant des solutions adaptatives pour contourner l'effet des politiques de contrôle d'accès. La réécriture de la requête est réalisée en utilisant l'architecture PSQRS qui effectue la prise de décision en se basant sur le modèle RBAC et la norme XACML. Notre solution de contrôle d'accès adaptatif est appliquée sur un cas d'utilisation de vidéo surveillance, tenant compte de la sensibilité du contenu consulté et de la mobilité des utilisateurs dans des contextes pervasifs. La solution proposée se situe dans une zone intermédiaire entre le respect de la rigidité des décisions d'accès et la flexibilité extrême de l'option «bris-de-glace » qui est souvent employée dans les situations critiques.

Comme perspective, nous envisageons d'étendre notre proposition en tenant compte d'autres éléments contextuels qui pourraient aussi influencer l'accessibilité aux contenus multimédias (e.g., matériels, réseau, bande passante) et d'appliquer le processus d'adaptation non seulement au niveau de la visualisation mais aussi au niveau du choix des algorithmes d'indexation explicites qui sont aussi protégés par les contraintes RBAC.

Bibliographie et références

Abreu B., Botelho L., Cavallaro A., Douxchamps D., Ebrahimi T., Figueiredo P., Macq B., Mory B., Nunes L., Orri J., Trigueiros M. J., Violante A., (2000) Video-Based Multi-Agent Traffic Surveillance System, *Actes du IEEE Intelligent Vehicles Symposium. IEEE*, p. 457-462

Agosti M., Buccio E. D., Nunzio G. M. D., Ferro N., Melucci M., Miotto R., Orio N., (2007) Distributed information retrieval and automatic identification of music works in SAPIR, *Actes du 15th Italian Symposium on Advanced Database Systems (SEBD)*, p. 479-482.

Al Kukhun D., Sedes, F. (2008): Adaptive Solutions for Access Control within Pervasive Healthcare Systems; *Actes du International Conference On Smart homes and health Telematics (ICOST 2008),* p.42-53.

Batko M., Falchi F., Lucchese C., Novak D., Perego R., Rabitti F., Sedmidubsky J., Zezula P. (2010). Building a web-scale image similarity search system. *Multimedia Tools Appl.* vol. 47, n. 3.

Berry M. W., Castellanos M., (2008). *Survey of Text Mining II: Clustering, Classification, and Retrieval*, Springer.

Bertino E., Bonatti P. A., Ferrari E., (2001) "TRBAC: A temporal role-based access control model," ACM Trans. Inf. Syst. Secur., vol. 4, no. 3, pp. 191–233.

Bertino E., Catania B., Damiani M. L., Perlasca P., (2005) "GEO-RBAC: a spatially aware RBAC," in 10th ACM Symposium on Access Control Models and Technologies SACMAT. ACM, pp. 29–37.

Boisson F., Crucianu M., Vodislav D., (2008). *Publication Framework for Content-Based Search in Heterogeneous Distributed Multimedia Databases*. Rapport de recherche CEDRIC n° 1585, 18 pages.

Brut M., Codreanu D., Dumitrescu S., Manzat A.-M., Sedes F. (2011): A distributed architecture for flexible multimedia management and retrieval. *Acte du Database and Expert Systems Applications (DEXA, 2011), p.*249-263

Brut M., Codreanu D., Manzat A.-M., Sèdes, F. (2011): Adapting Indexation to the Content, Context and Queries Characteristics in Distributed Multimedia Systems. *Acte du International Conference on Signal-Image Technology & Internet-Based Systems (SITIS 2011),* p.118-125.

Chen S.-C., Shyu M.-L., Zhao N. (2004): SMARXO: towards secured multimedia applications by adopting RBAC, XML and object-relational database. *Acte du 12th annual ACM international conf. on Multimedia,* p. 432-435.

El-Khoury, V. (2009): A Multi-level Access Control Scheme for Multimedia Database. *Acte du Workshop on Multimedia Metadata (WMM'09)*.

Ferraiolo D. F., Kuhn R. D., (1992) Role-Based Access Controls. *Acte du 15th National Computer Security Conference*, p.554-563.

Ferreira A., Chadwick D., Farinha P., Correia R., Zao G., Chilro R., Antunes L.,(2009). "How to securely break into RBAC: The BTG-RBAC Model," in Computer Security Applications Conference, ACSAC '09, pp. 23 –31.

Giroux P., Brunessaux S., Brunessaux S., Doucy J., Dupont G., Grilheres B., Mombrun Y., Saval A., (2008). Weblab : An integration infrastructure to ease the development of multimedia processing applications, *Actes du 21st Conference on Software and Systems Engineering and their Applications*.

Hansen F., Oleshchuk V. (2003), "SRBAC: A spatial role-based access control model for mobile systems," in Proceedings of the 7th Nordic Workshop on Secure IT Systems.

Harrison M. A., Ruzzo W. L., Ullman J. D.,(1976) "Protection in operating systems," *Commun. ACM*, vol. 19, no. 8, pp. 461–471.

Jaspers E.G.T., Wijnhoven R.G.J., Albers A.H.R., Desurmont X., Barais M., Hamaide J., Lienard B., (2005). CANDELA - Storage, Analysis and Retrieval of Video Content in Distributed Systems: Real-time Video Surveillance and Retrieval, *Actes du International Conference on Multimedia and Expo*, p. 1553 – 1556.

Joint NEMA/COCIR/JIRA Security and Privacy Committee (SPC), (2004), "Break-glass - an approach to granting emergency access to healthcare systems," White paper.

Kawagoe K., Kasai K., (2011). "Situation, team and role based access control," Journal of Computer Science, vol. 7, no. 5, pp. 629–637.

Kosch H., Maier P., (2009). Content based image retrieval systems – reviewing and benchmarking, *Actes du 9th Workshop on Multimedia Metadata*, p. 1-21.

Kuhn D.R. (1997). "Mutual Exclusion of Roles as a Means of Implementing Separation of Duty in Role-Based Access Control Systems". 2nd ACM Workshop Role-Based Access Control. P. 23–30.

National Institute of Standards and Technology (2006), "Assessment of access control systems," Interagency Report 7316.

OASIS (2003), A brief Introduction to XACML, http://www.oasis-open.org/committees/download.php/ 2713/Brief_Introduction_to_XACML.html

OASIS (2005), "Core and hierarchical role based access control (RBAC) profile of XACML v2.0". February 2005, http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-rbac-profile1-spec-os.pdf

Park S.-H., Han Y.-J., Chung T.-M., (2006), "Context-role based access control for context-aware application," in High Performance Computing and Communications, ser. Lecture Notes in Computer Science, Springer Berlin Heidelberg, vol. 4208, pp.572–580.

Petkovic M. , Jonker W., (2005) Content-Based Video Retrieval: A Database Perspective, *Multimedia Systems and Applications*, Berlin: Springer Verlag, vol. 25.

Pietarila P., Westermann U., Jarvinen S., Korva J., Lahti J., Lothman H., (2005). Candela-storage, analysis, and retrieval of video content in distributed systems: Personal mobile multimedia management, *Actes du IEEE International Conference on Multimedia and Expo (ICME)*, p. 1557-1560.

Pinquier J., André-Obrecht R., (2006). Audio Indexing: Primary Components Retrieval - Robust Classification in Audio Documents, *Multimedia Tools and Applications*, vol. 30, n. 3, p. 313-330.

Povey D., (1999), "Optimistic security: a new access control paradigm," in Proceedings of the workshop on New security paradigms, ser. NSPW '99. ACM, pp. 40–45.

Sánchez M., López G., Cánovas O., Sánchez J.-A., Gómez-Skarmeta A. F. (2006): An access control system for multimedia content distribution. *Acte du Third European conference on Public Key Infrastructure: theory and Practice* (EuroPKI 2006),p. 169-183.

Sandhu R., "Role Hierarchies and Constraints for Lattice-Based Access Controls", ESORICS 1996, p. 65-79.

Snoek C. G., Worring M., (2005). Multimodal video indexing: A review of the state of the art, *Multimedia Tools and Applications*, vol. 25, n. 1(January 2005), p. 5- 35.

Thuraisingham B., Lavee G., Bertino E., Fan J., Khan. L. (2006): Access control, confidentiality and privacy for video surveillance databases. *Acte du eleventh ACM symposium on Access control models and technologies* (SACMAT '06), p.1-10.

Viaud M.-L., Thièvre J., Goëau H., Saulnier A., Buisson O., (2008). Interactive components for visual exploration of multimedia archives, *Actes du 7th ACM International Conference on Image and Video Retrieval (CIVR)*, p. 609-616.

Zhang G., Parashar M., (2003). "Dynamic context-aware access control for grid applications," in Proceedings of the 4th International Workshop on Grid Computing, ser. GRID '03. IEEE Computer Society, pp. 101–108.