

5. Gestion des données et vie privée du citoyen urbain

- Architectures de gestion de données face au respect de la vie privée
- **Gestion de la vie privée dans les réseaux sociaux mobiles**
- Privacy-by-design: gestion de données confinée (puces et capteurs)
- Gestion de la vie privée dans les applications mobiles participatives
- Traitements de données globaux respectueux de la vie privée

Animesh Pathak

VILLES INTELLIGENTES : DÉFIS TECHNOLOGIQUES ET SOCIÉTAUX

Aujourd'hui, nous allons parler de la gestion de la vie privée dans les réseaux sociaux mobiles.

Les réseaux sociaux mobiles

- Villes intelligentes = citoyens connectés par téléphones mobiles
- Citoyens = vie sociale
- Applications web pour réseaux sociaux, mais aussi
- Les réseaux "mobile first" -> WhatsApp, SnapChat, Tinder

**Assurer la vie privée est
un enjeu majeur !**

2

Grâce aux citoyens qui sont connectés par les téléphones mobiles qui ont définitivement un rôle social, nous trouvons des applications mobiles pour les réseaux sociaux, par exemple Facebook.

Il y a maintenant également des réseaux sociaux qui sont principalement utilisés sur les mobiles. Par exemple Whatsapp, Snapchat et Tinder.

S'assurer du respect de la vie privée sur ces réseaux est un enjeu majeur.

La vie privée dans les réseaux sociaux mobiles

- Les téléphones stockent beaucoup d'informations personnelles
 - Nom
 - Adresse
 - Position actuelle
 - Photos
 - Lieux visités
 - Liste des "amis"

**Il faut contrôler qui peut
accéder à quelle information**

3

Le respect de la vie privée dans réseaux sociaux mobiles est primordial parce que les téléphones stockent beaucoup d'informations personnelles, entre autres, votre nom, adresse, votre position actuelle, vos photos, les lieux visités mais aussi la liste de vos amis.

Il faut donc **contrôler qui peut accéder à quelle info**. Concrètement, il faut un système de contrôle d'accès aux informations relevant de la sphère privée.

Un contrôle d'accès idéal

- **Expressif**

- "Tout le monde peut lire l'adresse email de Bob"
- "Les amis de Bob peuvent lire l'adresse email et postale de Bob"

4

Un système de contrôle d'accès idéal aura les propriétés suivantes :

- Il doit d'abord être **expressif**. Je peux par exemple dire « Tout le monde peut lire l'adresse e---mail de Bob » ou « Seuls les amis de Bob peuvent lire l'adresse e---mail et postale de Bob. »

Un contrôle d'accès idéal

- Expressif
- **Flexible**

- Les préférences pour le contrôle d'accès sont très subjectives :
 - ✓ "Les amis d'Alice peuvent accéder aux données de Bob"

5

- Il doit être **flexible** ou encore, disons que l'on peut décrire des contraintes très subjectives. Par exemple, « Les amis d'Alice peuvent accéder aux données de Bob. » De telles contraintes comme nécessitent un système flexible.

Un contrôle d'accès idéal

- Expressif
- Flexible
- **Un modèle sensible aux liens sociaux**
 - Défini en terme de concepts sociaux
 - ✓ Distingue les amis et les collègues de Bob
 - ✓ Peut s'adapter à l'évolution de l'environnement social

6

- La troisième caractéristique est un modèle **sensible aux liens sociaux**. Il s'agit d'offrir un modèle qui permet de définir des règles d'accès en termes de concepts sociaux, par exemple, qui peut distinguer entre les amis et collègues de Bob et qui peut aussi évoluer s'il y a un collègue qui n'est plus un collègue de Bob.

Un contrôle d'accès idéal

- Expressif
- Flexible
- Un modèle sensible aux liens sociaux
- **Capable de fonctionner sur différentes plateformes**
 - Smart Phone
 - PC Portable
 - Appli "Cloud"

7

- Un système de contrôle d'accès idéal doit aussi être **capable de fonctionner sur différentes plateformes**. Par exemple les smartphones, les PC portables, les applis stockées dans le cloud.

Un contrôle d'accès idéal

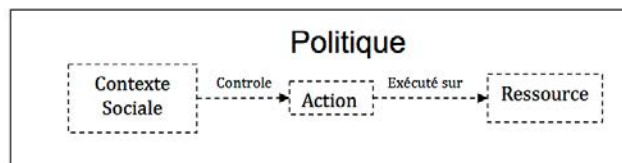
- Expressif
- Flexible
- Un modèle sensible aux liens sociaux
- Capable de fonctionner sur différentes plateformes
- **Intuitif et facile à utiliser**
 - Pas difficile de s'exprimer
 - Même pour les non-informaticiens !

8

- Finalement, le système doit être **intuitif et facile à utiliser**. Il doit être très facile d'exprimer les "politiques" même pour les non-informaticiens.

Modèle de Politique

- 3 composants
 - Ressource : Objet à accéder
 - Contexte Social : conditions qui contrôlent l'accès aux ressources, liées à la condition sociale de l'objet
 - Action : l'action à exécuter sur une ressource



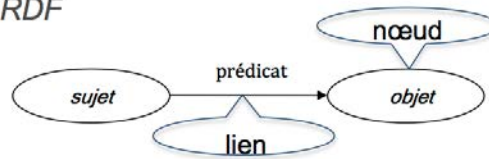
9

Le modèle de politique dont nous allons parler dans cette séquence est constitué de 3 composants :

- Le premier est la **ressource** : l'objet à accéder,
- Le deuxième est le **contexte social** : ce sont les conditions qui contrôlent l'accès aux ressources et qui sont liées à la condition sociale de l'objet,
- Le troisième est **l'action à exécuter sur les ressources** : lecture, écriture, le fait d'effacer des choses par exemple.

1. Ressources

- Les ressources sont les données sociales de chaque utilisateur
- Nous utilisons le modèle défini dans [TONINELLI2010]
- Les données sociales sont représentées en « Resource Description Framework » (RDF) :
- Phrases RDF:
 - < sujet, prédicat, objet >
- Triplets RDF liés ensemble
 - dans un graphe RDF



10

Les ressources sont les données sociales de chaque utilisateur.

Nous utilisons le modèle défini dans l'article cité en référence [TONINELLI2010].

Les données sociales y sont représentées au format RDF (pour « Resource Description Framework »), où l'information est toujours représentée par des phrases.

Une phrase consiste en un sujet, un prédicat et un objet, ce qui constitue un « triplet ».

Et les différents triplets RDF sont liés ensemble dans un graphe RDF.

2. Actions et 3. Contexte Social

- 3 types d'actions :
 - Read : lecture de triplet(s)
 - Add : ajout de triplet(s)
 - Remove : suppression de triplet(s)
- Contexte Social : l'information sociale qui permet de contrôler l'accès à une ressource.
 - "Les amis de Bob"
 - "Les collègues de Bob"
 - "Les sujets qui intéressent Bob"

11

Il y a 3 types d'actions pour une plateforme de contrôle d'accès : lecture, ajouter des triplets ou supprimer des triplets.

Pour le contexte social, c'est l'information sociale qui permet de contrôler l'accès à une ressource. Par exemple : « Qui sont les amis de Bob? Les collègues de Bob ? Ou les sujets qui intéressent Bob?»

Représentation Formelle de Politiques (1)

- Politiques définies comme des requêtes sur des données RDF

- Langage de requête SPARQL:

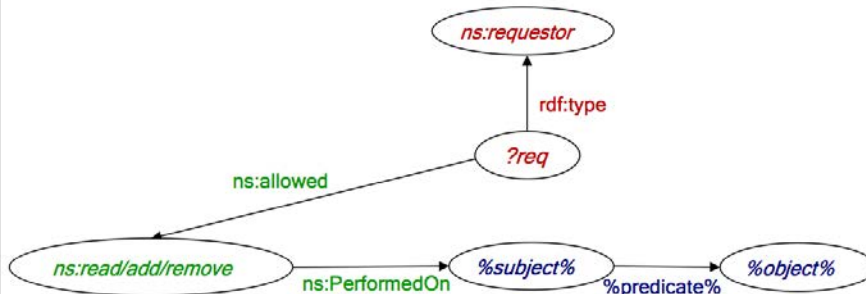
```
CONSTRUCT {%subject% %predicate% %object%.}  
WHERE{?req rdf:type ns:requestor.  
      ?req ns:allowed ns:(read/add/remove).  
      ns:(read/add/delete) ns:PerformedOn subject%.  
      %subject% %predicate% %object%.}
```

12

Une représentation formelle des politiques est produite au moyen du langage de requête SPARQL où l'on voit que l'on peut construire un triplet avec les variables sujet, prédicat et objet et où l'on peut identifier quelqu'un qui veut faire une demande et indiquer une permission de faire une action s'il y a un contexte spécial.

Représentation Formelle de Politiques (2)

- Représentation graphique d'une politique



13

Ici, nous voyons la représentation graphique d'une politique.

On voit que quelqu'un est un « requestor », ou personne qui a fait une demande, il est autorisé à faire une action sur une partie d'un graph d'une certaine forme.

Exemple : Lecture de triplets

- Requête
 - « Bob veut lire la liste des amis d'Alice qui sont membres d'Inria »
- Requête « Read »
 - Envoyée comme un pattern SPARQL
<Alice mse:knows ?obj.>
- Politique
 - Chaque membre d'Inria peut lire la liste des amis d'Alice qui sont aussi membres d'Inria

14

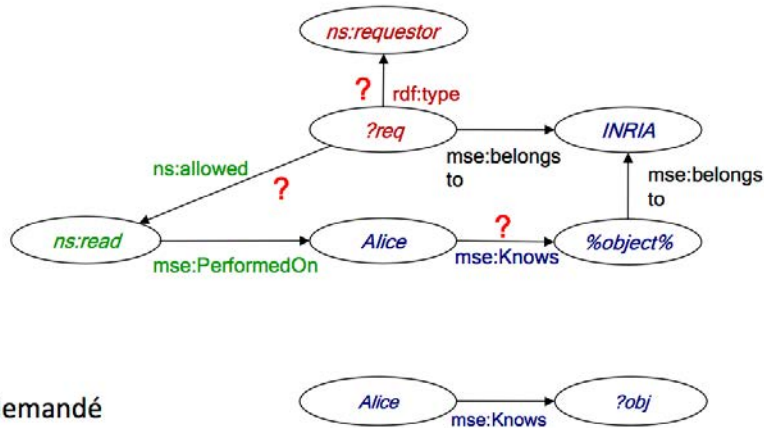
La **requête** est la suivante : « Bob veut lire la liste des amis d'Alice qui sont membres d'Inria. »

On voit que c'est une action de lecture. Elle est envoyée comme un pattern SPARQL où le sujet est Alice, le prédicat est « knows » ou « connaît », et l'objet n'est pas défini : <Alice mse:knows ?obj.>

On peut supposer que la **politique** est la suivante : chaque membre d'Inria peut lire la liste des amis d'Alice qui sont aussi membres d'Inria. C'est-à dire que si quelqu'un n'est pas membre d'Inria, il ne peut pas lire la liste des amis d'Alice qui sont membres d'Inria.

Exemple : Lecture de triplets

Chaque membre d'Inria peut lire la liste des amis d'Alice qui sont aussi membres d'Inria.



15

Voyons comment cela se passe.

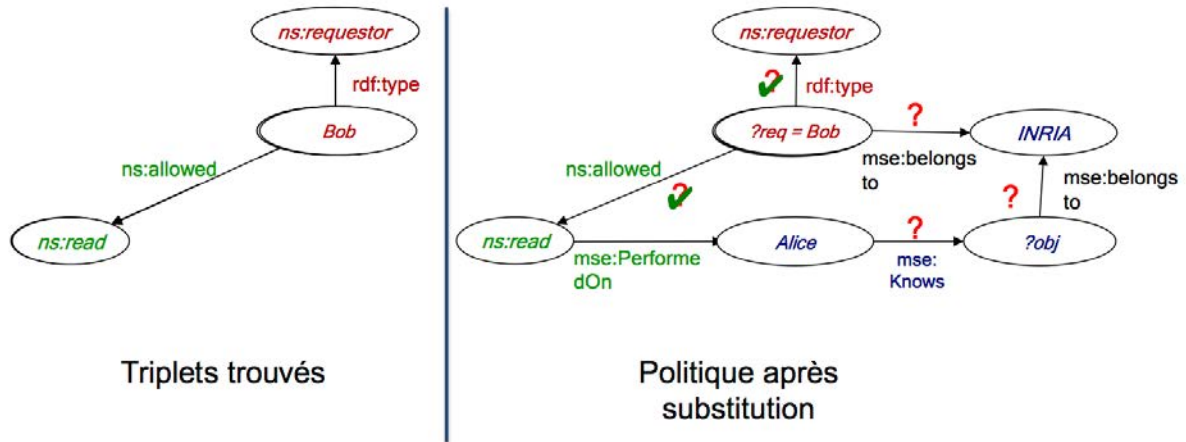
Tout d'abord, on peut représenter cette politique de la façon suivante : on voit qu'il doit y avoir un lien entre le noeud « demandeur » (requestor) et Inria, un lien de type « belongs to », signifiant que le demandeur est membre d'Inria.

Et c'est seulement si ce lien existe que le demandeur peut lire l'information des autres personnes qui sont aussi membres d'Inria.

Les triplets demandés auront la forme « Alice connaît quelqu'un. » On voit qu'Alice peut connaître une personne, et si cette personne est aussi membre d'Inria, l'accès sera autorisé.

Exemple de lecture : Requête et politique (1)

« Bob veut lire la liste des amis d'Alice qui sont membres d'Inria »

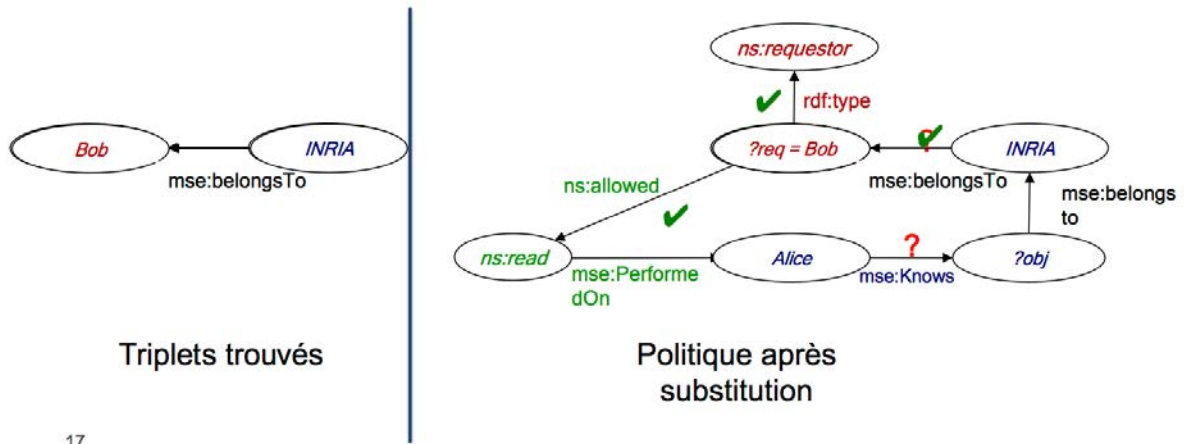


16

Dans l'exemple, on peut voir que Bob veut lire. Si on substitue les triplets qui sont déjà dans le système dans la politique, on a une partie des politiques qui est bien remplie maintenant.

Exemple de lecture : Requête et politique (2)

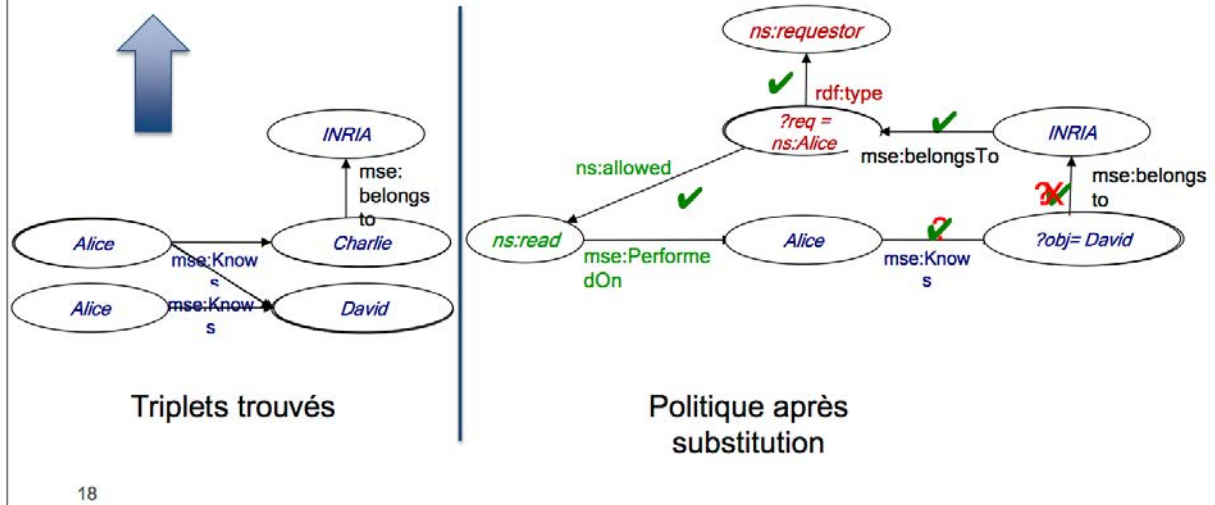
- « Bob veut lire la liste des amis d'Alice qui sont membres d'Inria »
- Condition : le demandeur doit être membre d'Inria



La condition de cette lecture est que le demandeur doit être membre d'Inria. On voit que Bob est bien membre d'Inria. Par conséquent, les conditions de la politique sont bien remplies.

Exemple de lecture : Exécution de la requête

- « Bob veut lire la liste des amis d'Alice qui sont membres d'Inria »



La toute dernière partie de contexte à remplir est que Bob peut lire seulement l'information des personnes qui sont membres d'Inria. On voit à gauche qu'Alice a 2 amis : Charlie qui est membre d'Inria et David qui ne l'est pas. Seule l'information de Charlie doit être rendue au demandeur, Bob, mais pas celle de David, parce que Charlie est membre d'Inria mais David ne l'est pas.

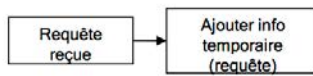
Vue d'ensemble de la vérification des Politiques

Requête
reçue

19

Maintenant, voyons ensemble le processus de vérification de la politique.
On commence avec une requête reçue.

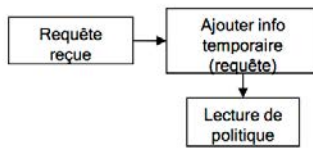
Vue d'ensemble de la vérification des Politiques



20

Tout d'abord, on ajoute l'information de la requête dans la base des connaissances.

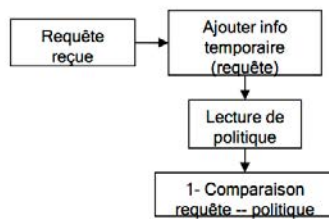
Vue d'ensemble de la vérification des Politiques



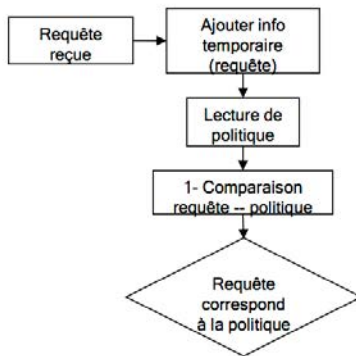
21

Ensuite, on lit la politique et on fait une comparaison entre la requête et la politique.

Vue d'ensemble de la vérification des Politiques



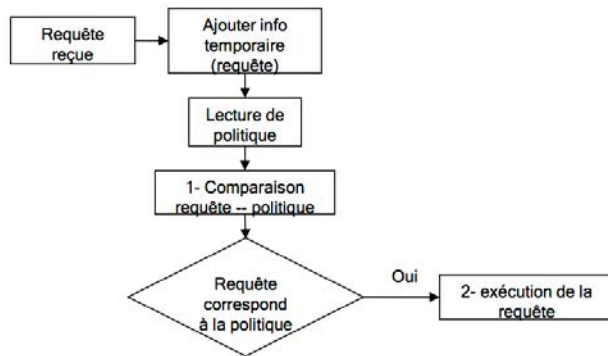
Vue d'ensemble de la vérification des Politiques



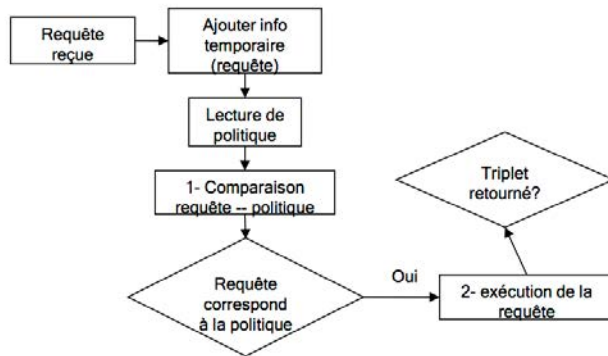
23

S'il y a des requêtes qui correspondent à la politique, on voit s'il y a des résultats.

Vue d'ensemble de la vérification des Politiques



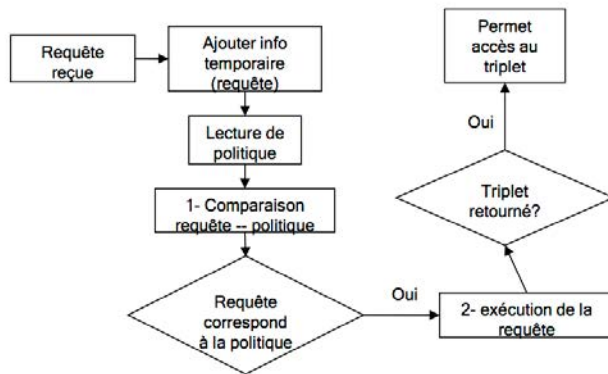
Vue d'ensemble de la vérification des Politiques



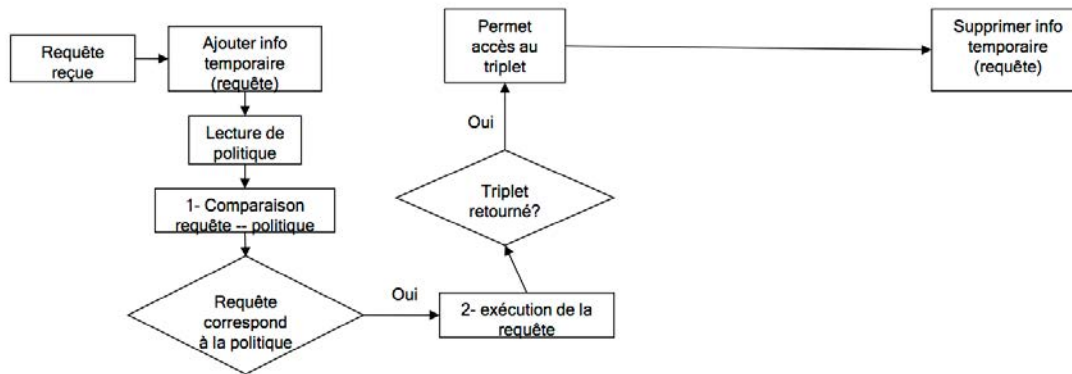
25

Si des triplets sont rendus, on autorise l'accès et on supprime l'information (requête reçue) qui a été ajoutée auparavant dans les requêtes.

Vue d'ensemble de la vérification des Politiques

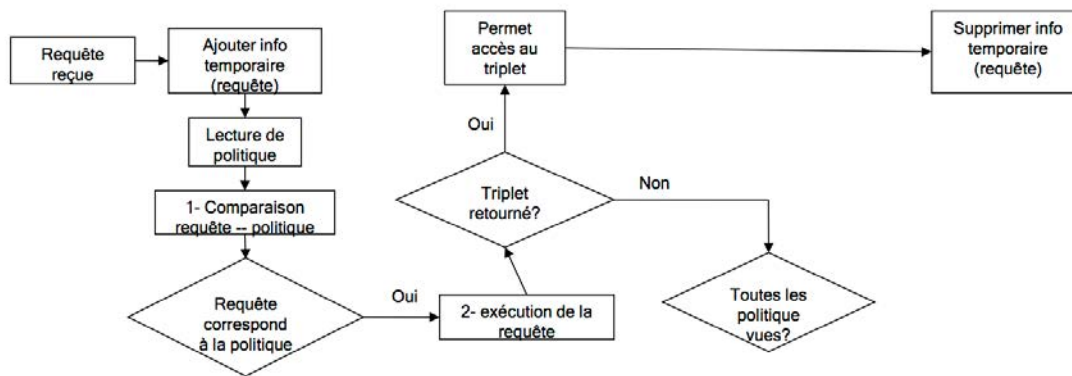


Vue d'ensemble de la vérification des Politiques



27

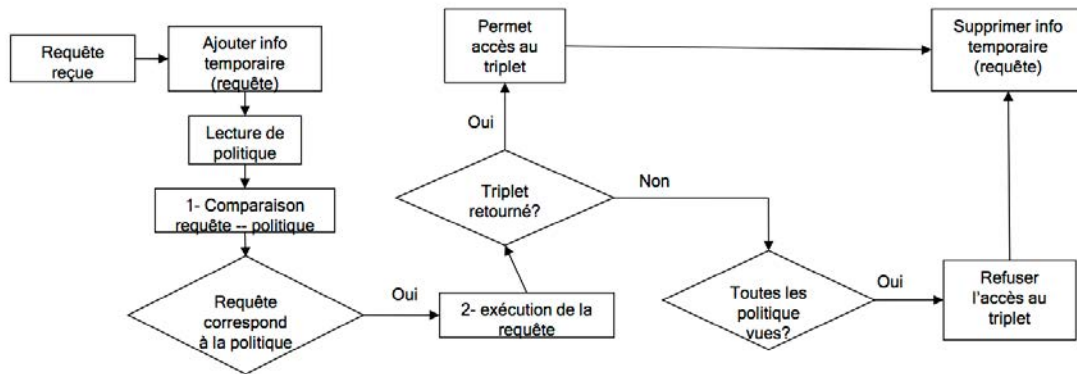
Vue d'ensemble de la vérification des Politiques



28

S'il n'y a pas de triplets rendus, on voit si toutes les politiques ont été utilisées.

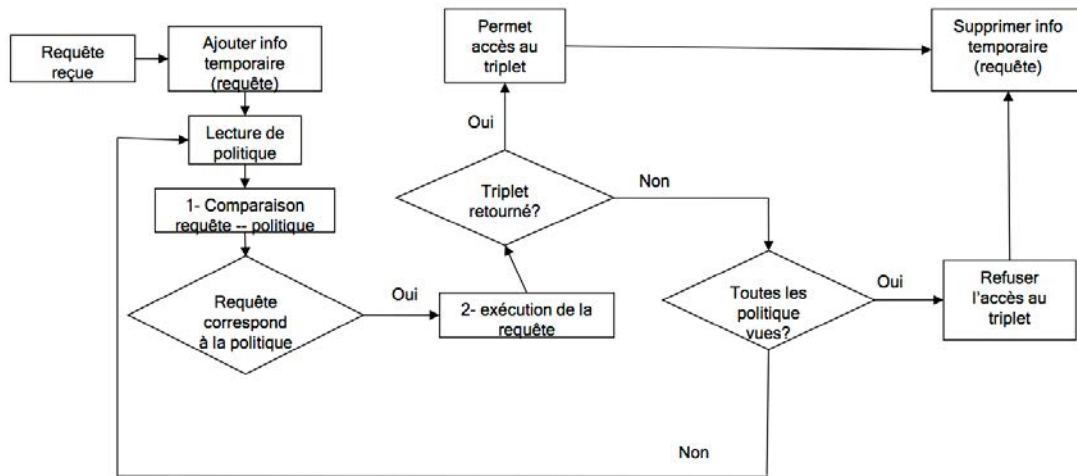
Vue d'ensemble de la vérification des Politiques



29

Si oui, on termine le processus,

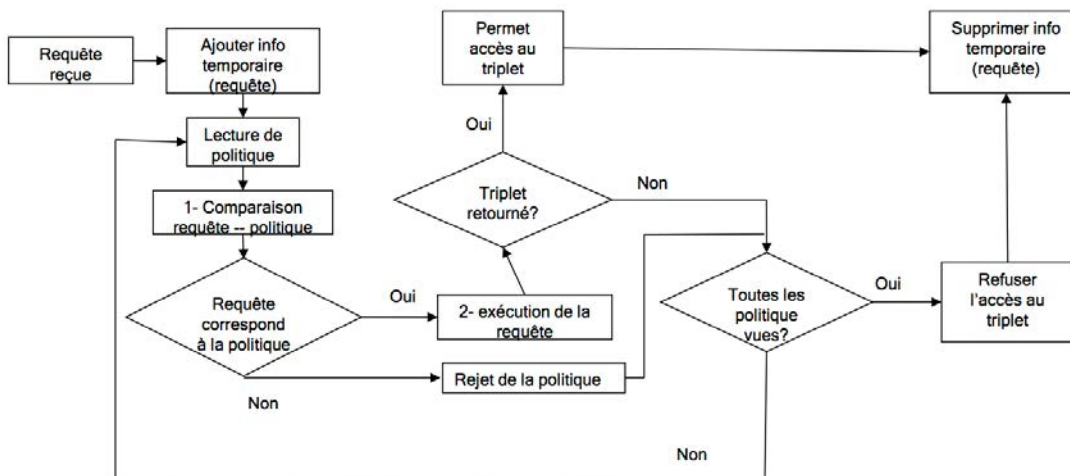
Vue d'ensemble de la vérification des Politiques



30

sinon on lit les autres politiques restantes.

Vue d'ensemble de la vérification des Politiques



31

Bien sûr, si les requêtes ne correspondent pas aux politiques, on rejette la politique immédiatement.

Conclusions

- La vie privée est extrêmement importante dans les réseaux sociaux mobiles
- Un système de contrôle d'accès idéal doit être :
 - expressif, flexible, et sensible aux aspects sociaux
 - facile à utiliser et exécutable sur les différentes plateformes
- Les techniques sémantiques constituent une base solide pour un système assurant le respect de la vie privée et le contrôle d'accès
- Il reste du travail, notamment en termes de performance et d'utilisation

32

Nous avons vu que la **vie privée** est **extrêmement importante dans les réseaux sociaux mobiles**.

Un **système de contrôle d'accès idéal** doit être **flexible, expressif et sensible aux aspects sociaux**, mais aussi **facile à utiliser et exécutable sur différentes plateformes**.

Nous avons discuté les **techniques sémantiques** qui constituent une **base solide** pour un système assurant le respect de la vie privée et les contrôles d'accès. On a vu comment cela se passe en utilisant un exemple.

Références

- [TONINELLI2010] Alessandra Toninelli, Animesh Pathak, Amir Seyedi, Roberto Speicys Cardoso, Valérie Issarny, "Middleware Support for Mobile Social Ecosystems", 2010, Proceedings of the 2nd IEEE International Workshop on Middleware Engineering, to be held with COMPSAC 2010.
- Sara Hachem, Alessandra Toninelli, Animesh Pathak, Valérie Issarny. Policy-based Access contrôle in Mobile Social Ecosystems. *IEEE International Symposium on Policies for Distributed Systems and Networks*, Jun 2011, Pisa, Italy. IEEE computer society, 2011.
- RDF : www.w3.org/RDF/
- SPARQL : <http://www.w3.org/TR/rdf-sparql-query>