Composition de services inter-domaines protégés par des modèles hétérogènes de contrôle d'accès

(titre provisoire)

Abdramane BAH[1], Christian ATTIOGBE[1], Pascal ANDRE[1], Jacqueline KONATE[2]

[1] LS2N - Université de Nantes(Equipe AELOS)
[2] Université des Sciences, des Techniques et des Technologies de Bamako (USTTB)

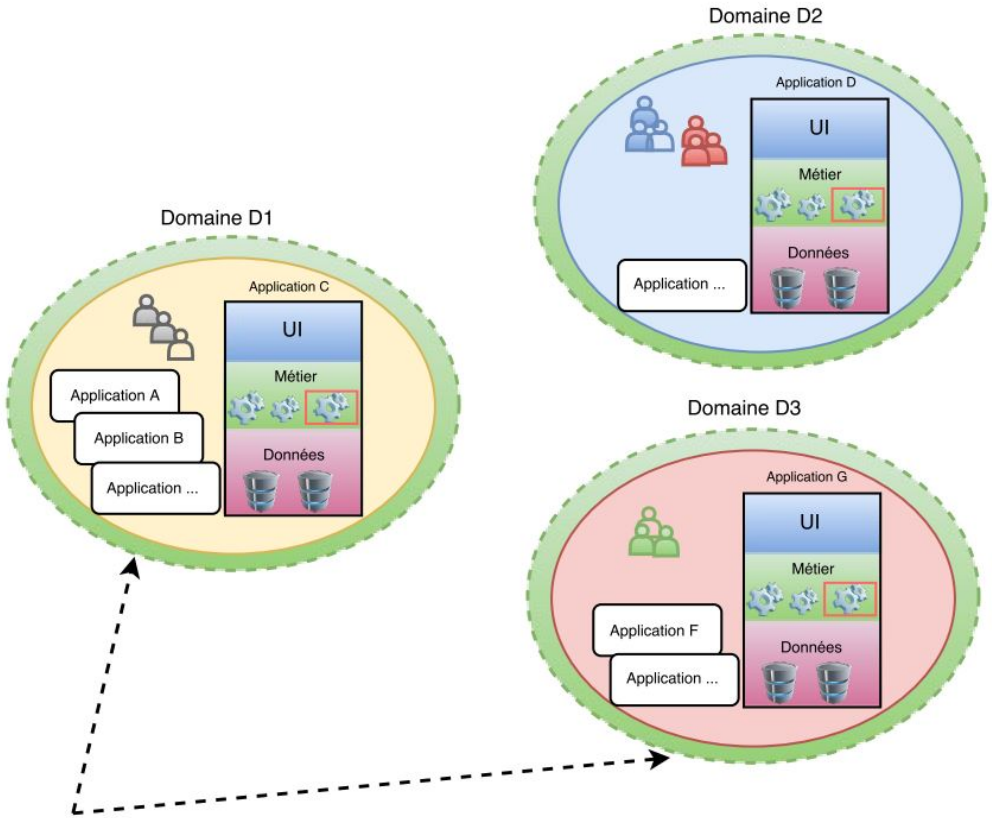
22 Mars 2018

Sommaire

Introduction

Collaboration entre domaines

On a des domaines qui souhaitent travailler ensemble



1. Domaines autonomes

La collaboration s'effectue via les systèmes d'information (SI)

Objectifs :
Productivité, efficacité, réduction des coûts

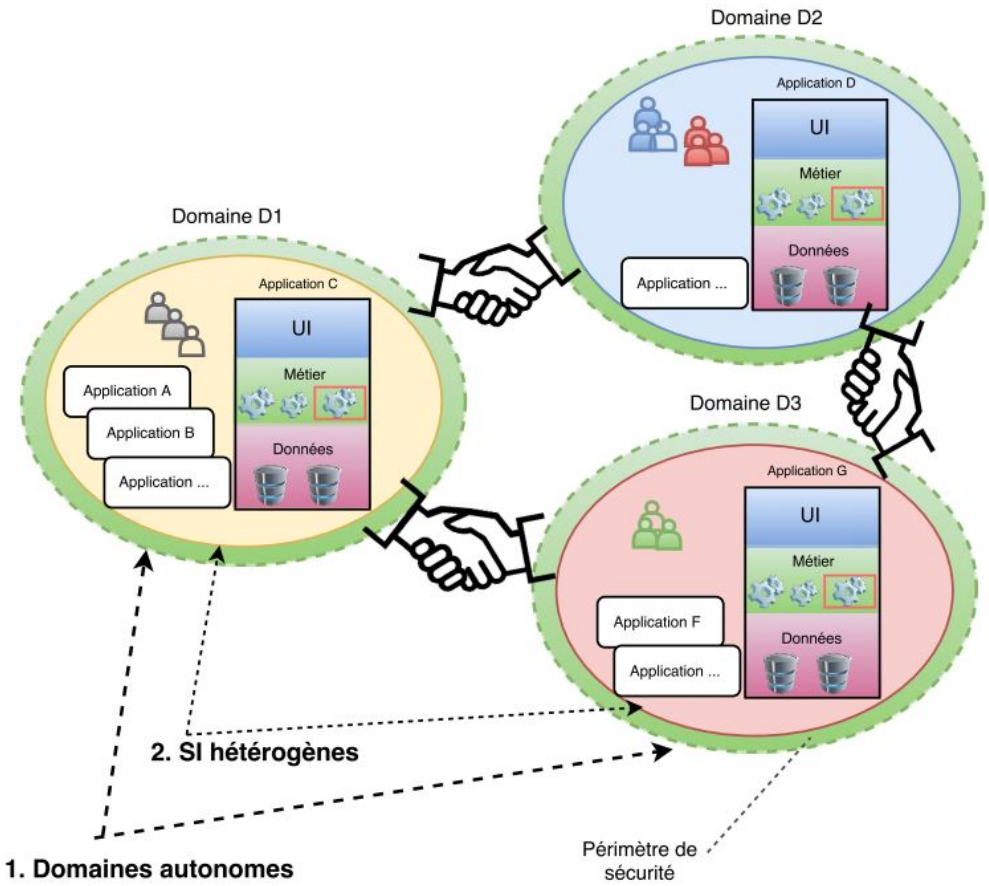
Fonctionnement :
Partager les ressources des uns avec les utilisateurs des autres

Domaine = Organisation, unité organisationnelle

ressources = Données, applications (fonctionnalités)

Introduction

Collaboration entre domaines



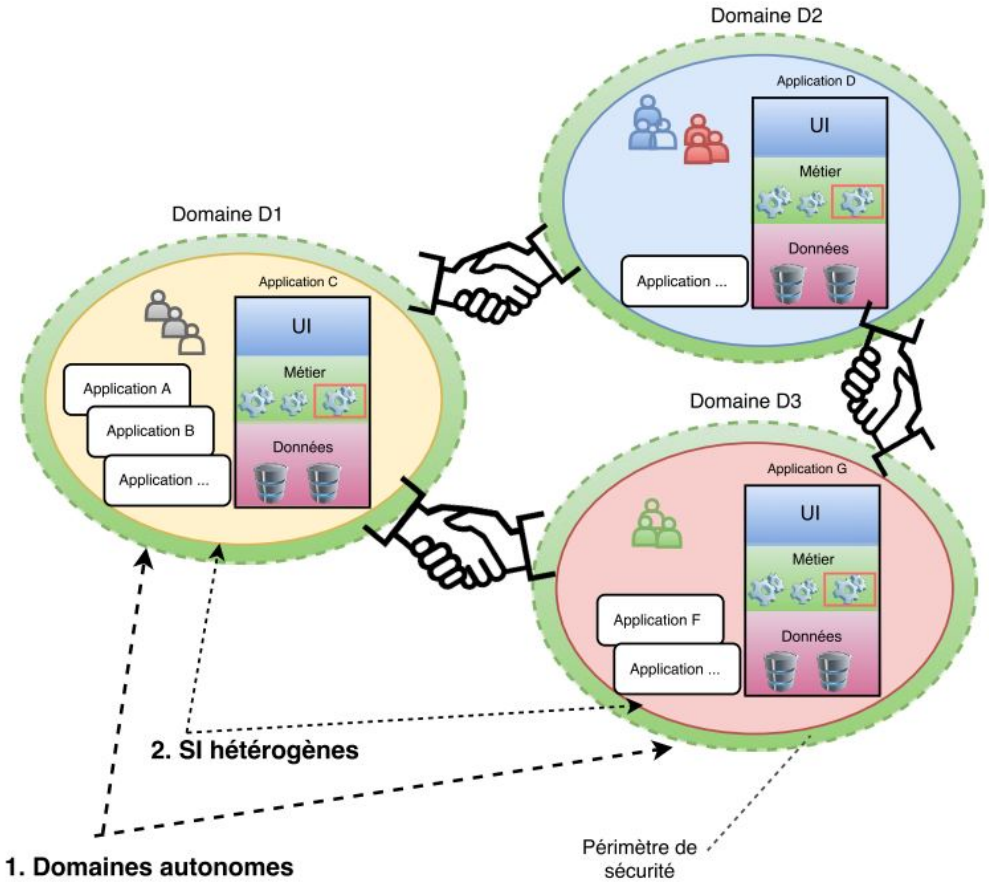
Chaque **domaine** :

- est géré par un système d'information (SI) composé d'applications hétérogènes
- est délimité par un périmètre de sécurité
- possède ses propres registres d'utilisateurs

Les SI hétérogènes des domaines doivent interagir

Introduction

Collaboration entre domaines



Comment faire interagir des SI hétérogènes ?

Réponse : les SI doivent être **interopérables**

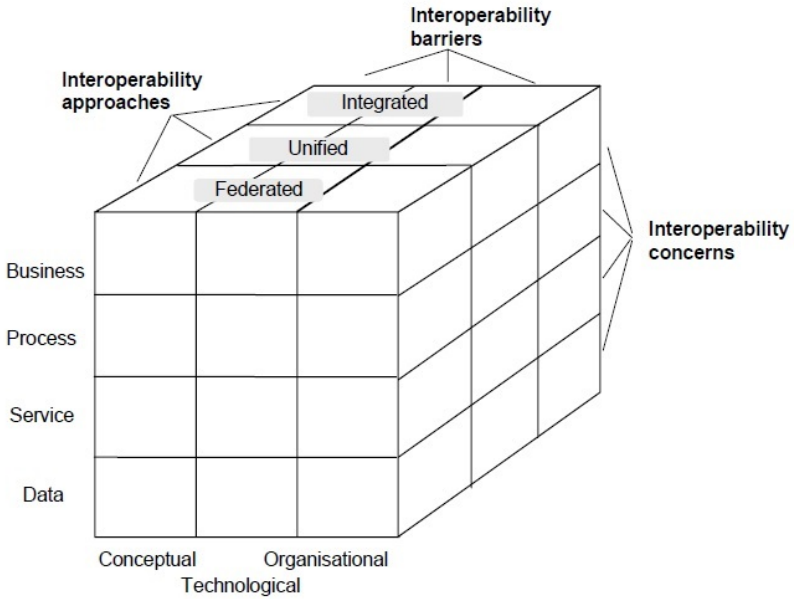
Pré-condition : Un utilisateur n'existe que dans un seul domaine

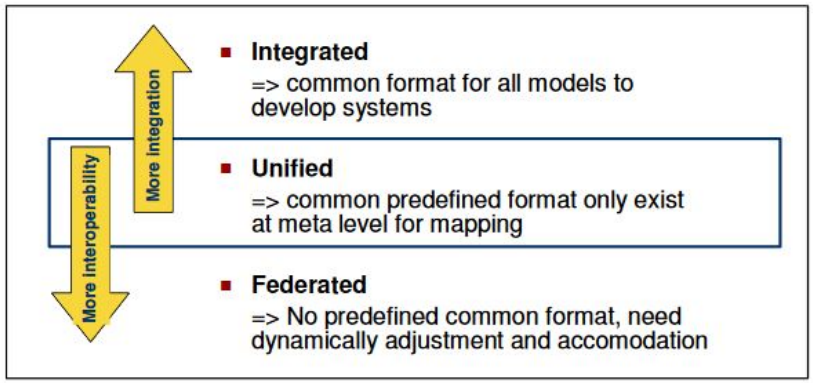
Comment rendre les SI interopérables ?

Sécurité des applications interopérables

Interopérabilité

C'est la capacité de deux ou plusieurs systèmes à communiquer et échanger des informations, utiliser les informations échangées et accéder aux fonctionnalités d'un tiers système [1]





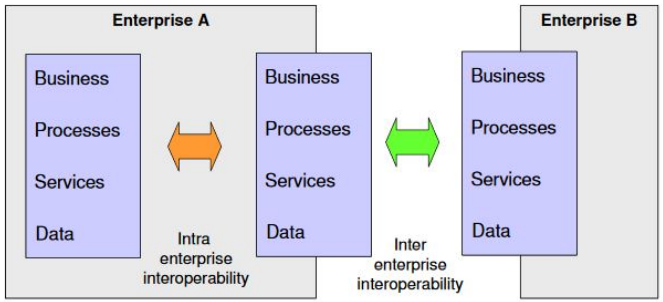
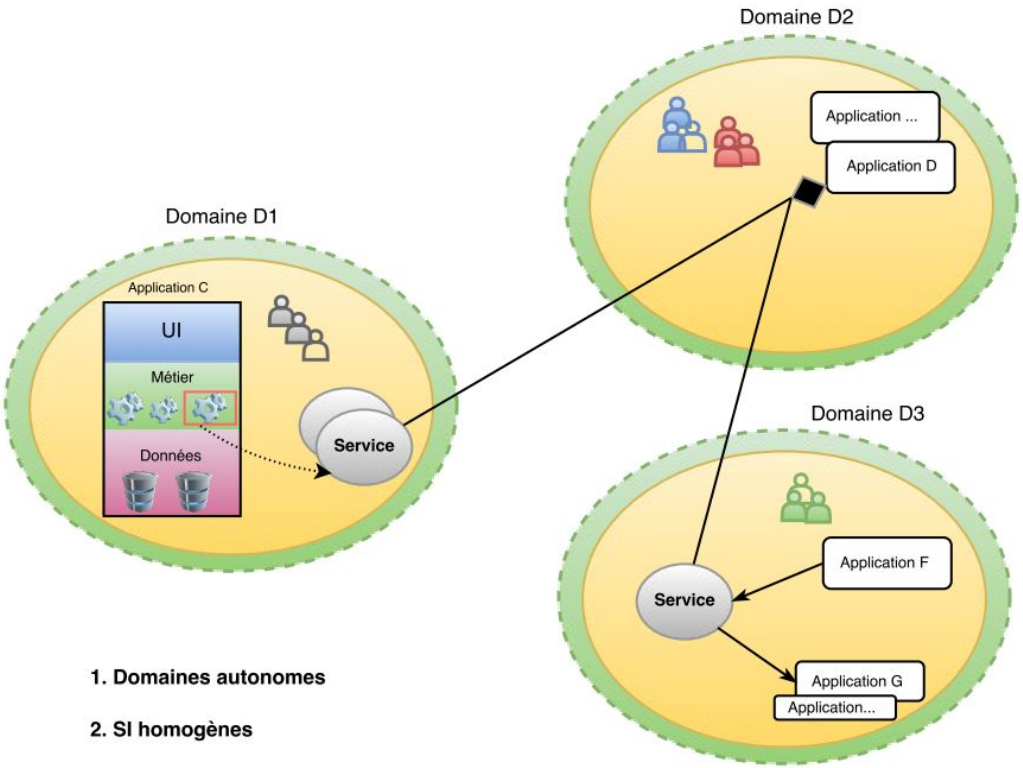
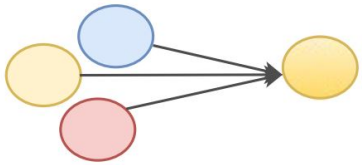


FIGURE – Framework d'interopérabilité d'entreprise [1]

Une solution à l'interopérabilité : **Architecture orientée service** (SOA)



Homogénéisation des SI



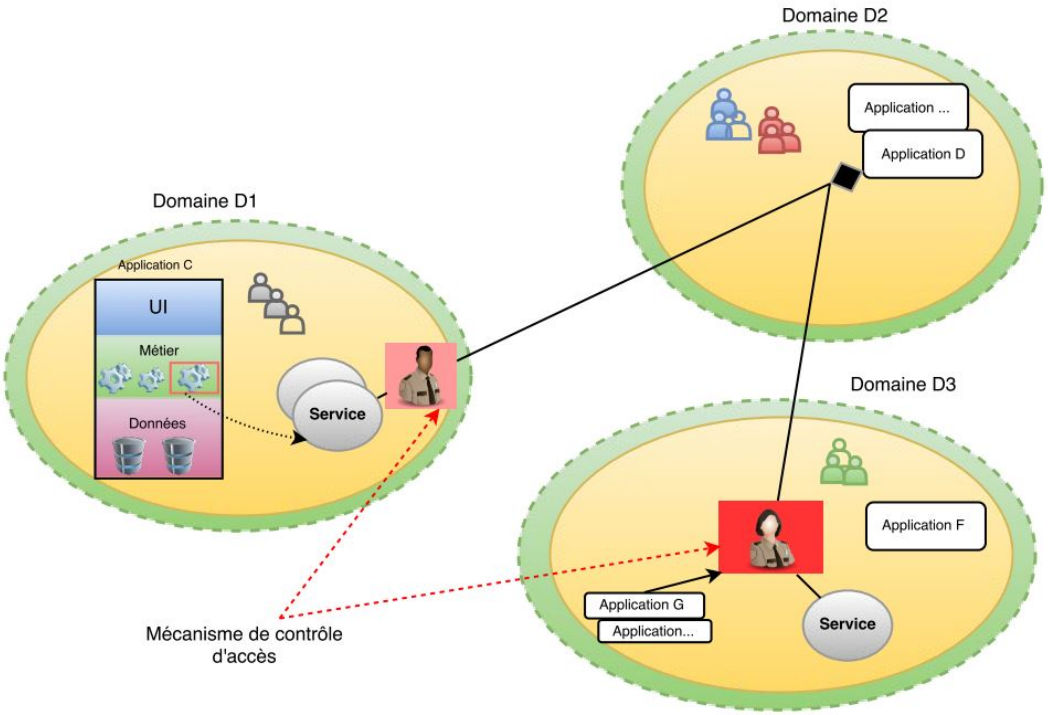
Cacher l'hétérogénéité des SI via les **services**

- Utilisation des standards de communication et d'échange d'information
- Utilisation de contrats standards bien définis
- Couplage faible, réutilisation

SOA permet l'interopérabilité entre des SI hétérogènes

Architecture orientée service : sécurité des services

Contrôle d'autorisations : L'accès aux services inter-domaines peut être contrôlé



Contrôle d'accès

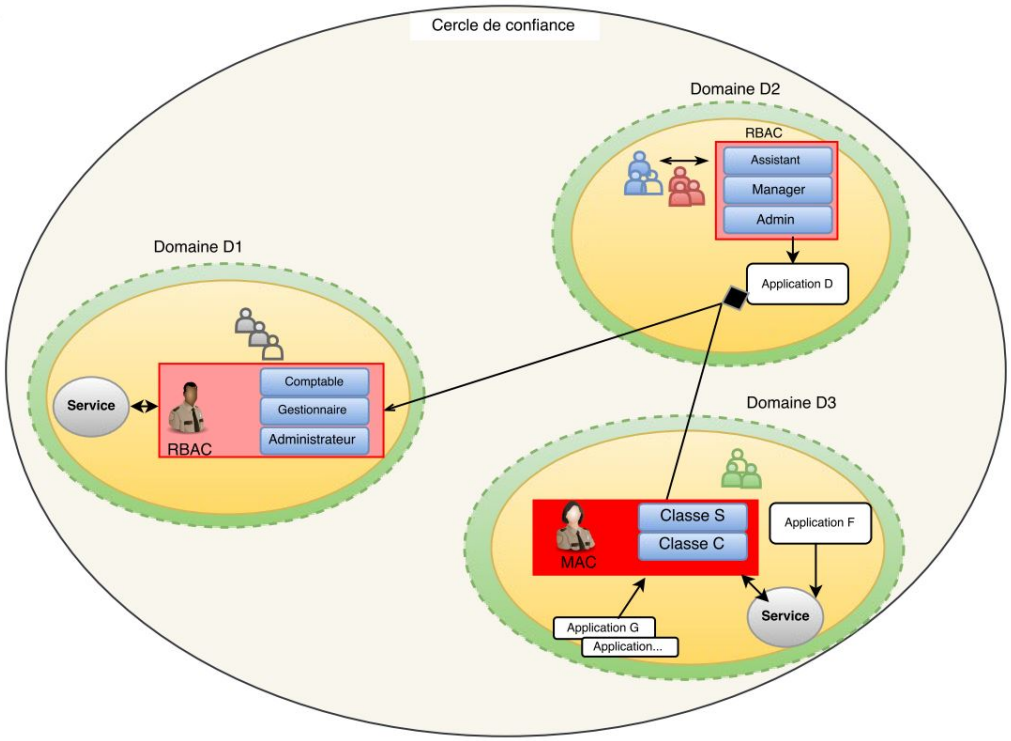
1. authentification des utilisateurs
2. Droits d'accès aux services

Rappel (**pré-condition**) :
Un utilisateur n'existe que dans un seul domaine

Comment authentifier un utilisateur qui n'existe pas dans mon domaine ?

Architecture orientée service : sécurité des services

Authentification (Fédération d'identité)



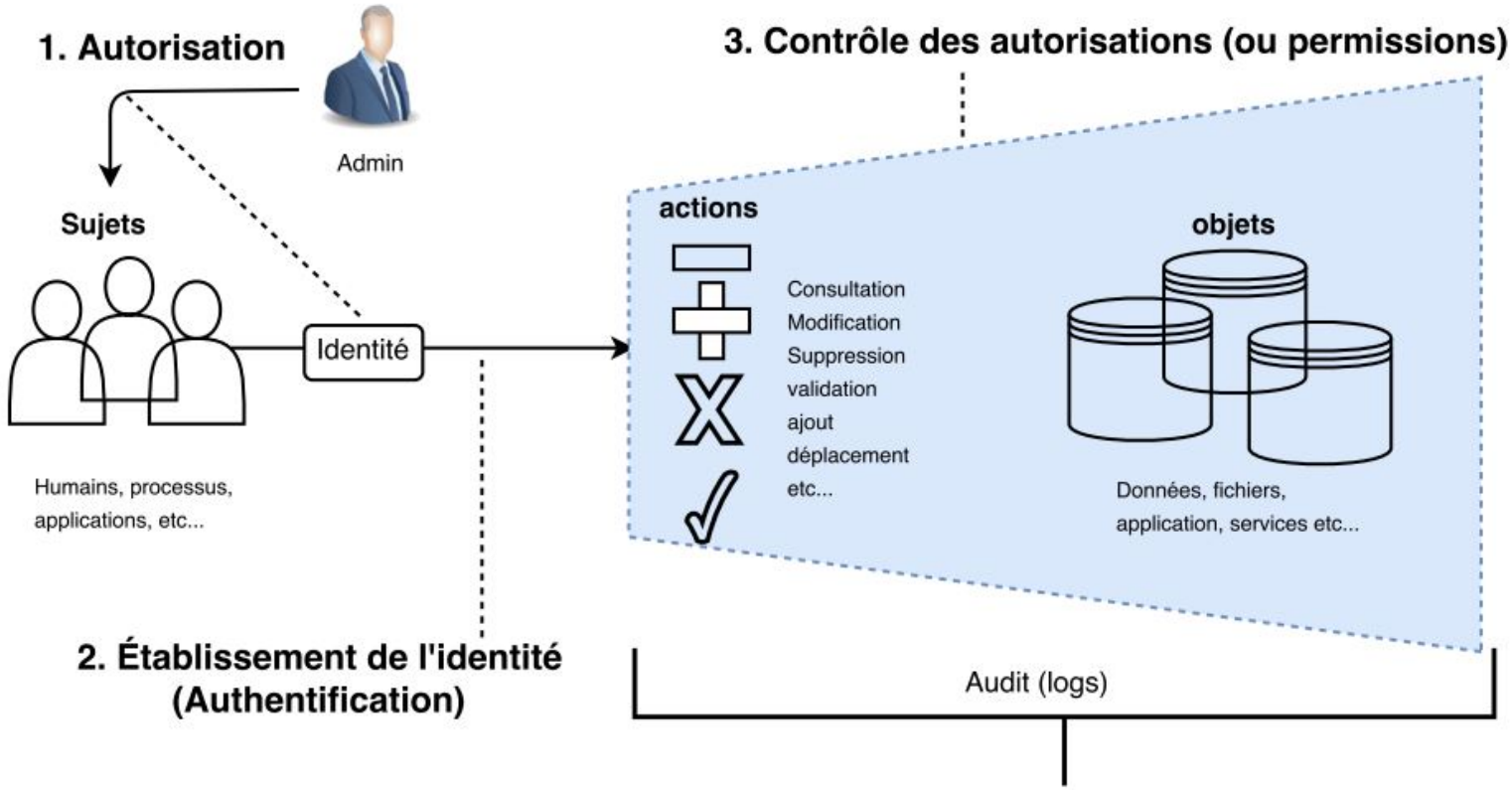
Confiance (*Trust*) :

- les domaines font confiance aux utilisateurs authentifiés des autres
- L'utilisateur s'authentifie une seule fois (SSO)
- L'utilisateur peut alors accéder aux différents services en utilisant son identité d'origine

Oui, la confiance règne maintenant,
Comment contrôler les accès des utilisateurs externes à mes services ?

Contrôle d'accès

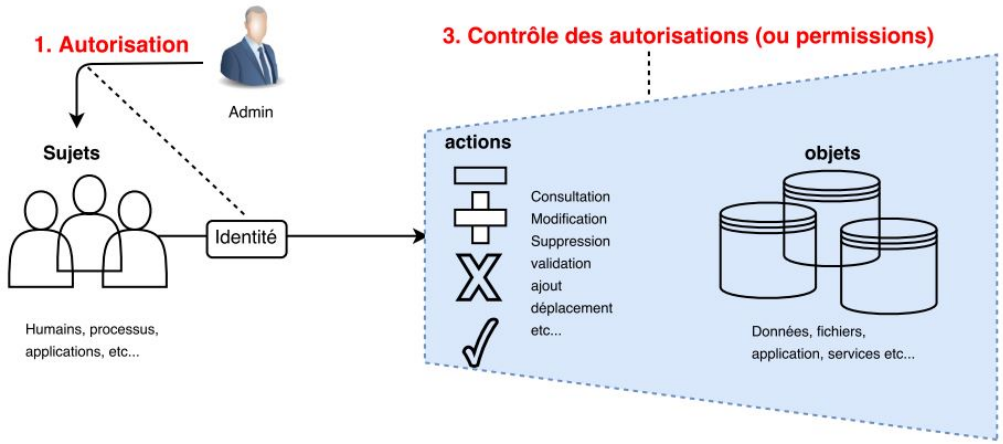
Ensemble de moyens permettant d'autoriser, de refuser ou de restreindre les actions des sujets uniquement aux objets auxquels ils ont droit.

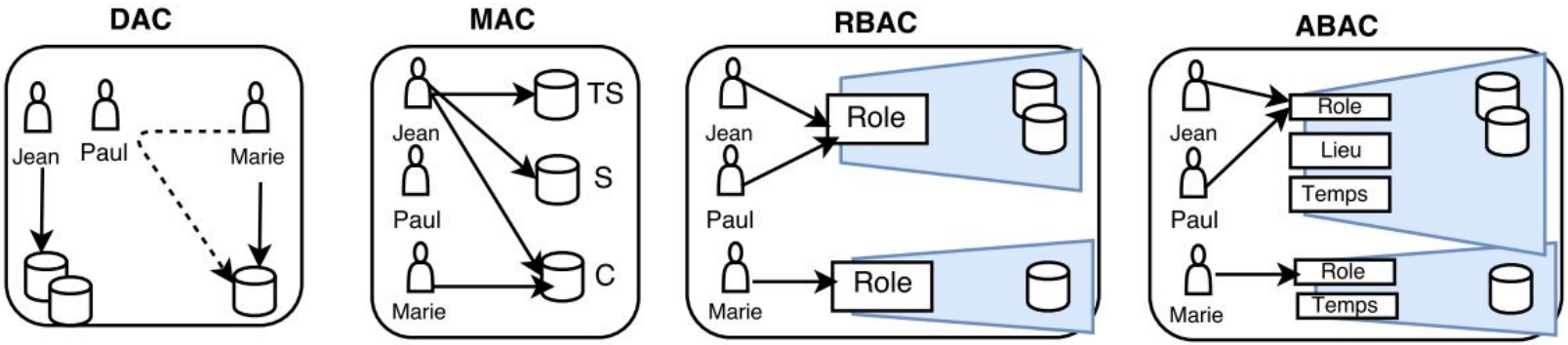


Contrôle d'accès

Phase (1) et (3) : les modèles de contrôle d'accès

le contrôle d'accès repose sur quatre classes de modèle d'autorisation

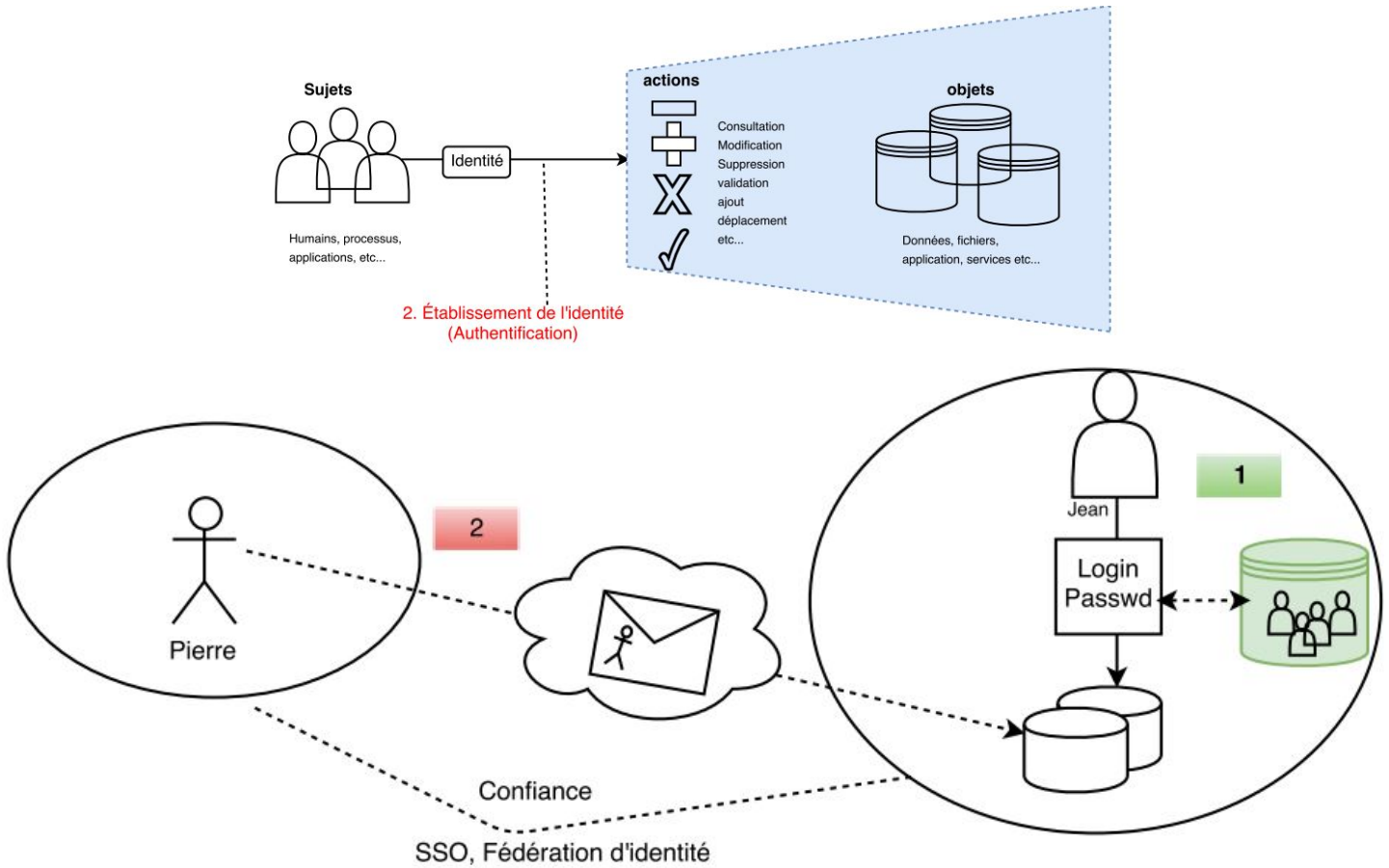




Contrôle d'accès

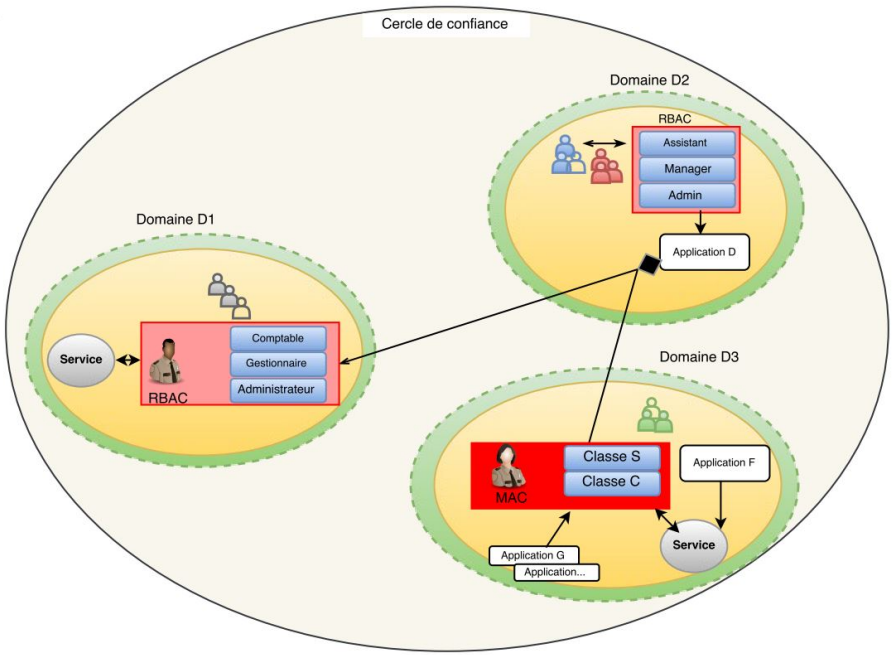
Phase (2) : Établissement de l'identité (Authentification)

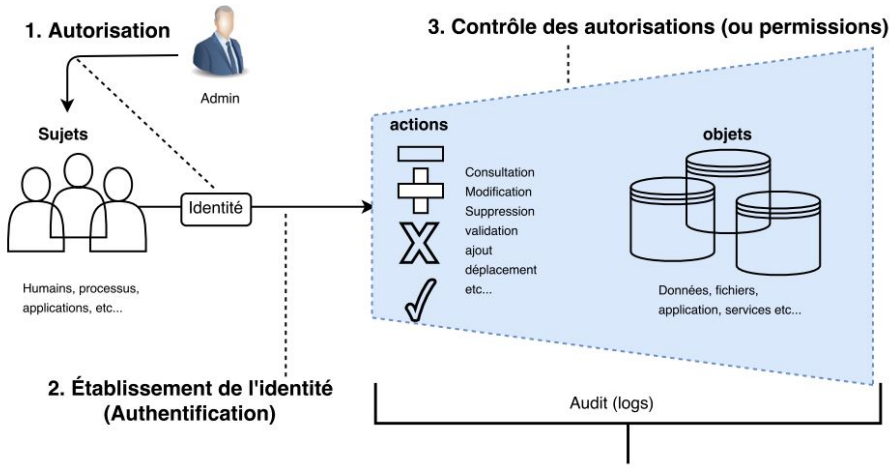
Deux cas : (a) Identités intra-domaine (b) Identités fédérées (Utilisateurs externes)



Contrôle d'accès

Autorisation des utilisateurs externes





- Un utilisateur n'existe que dans un seul domaine
- L'identité de l'utilisateur authentifié est propagée avec sa requête aux services
- L'utilisateur externe doit acquérir des droits d'accès à la volée

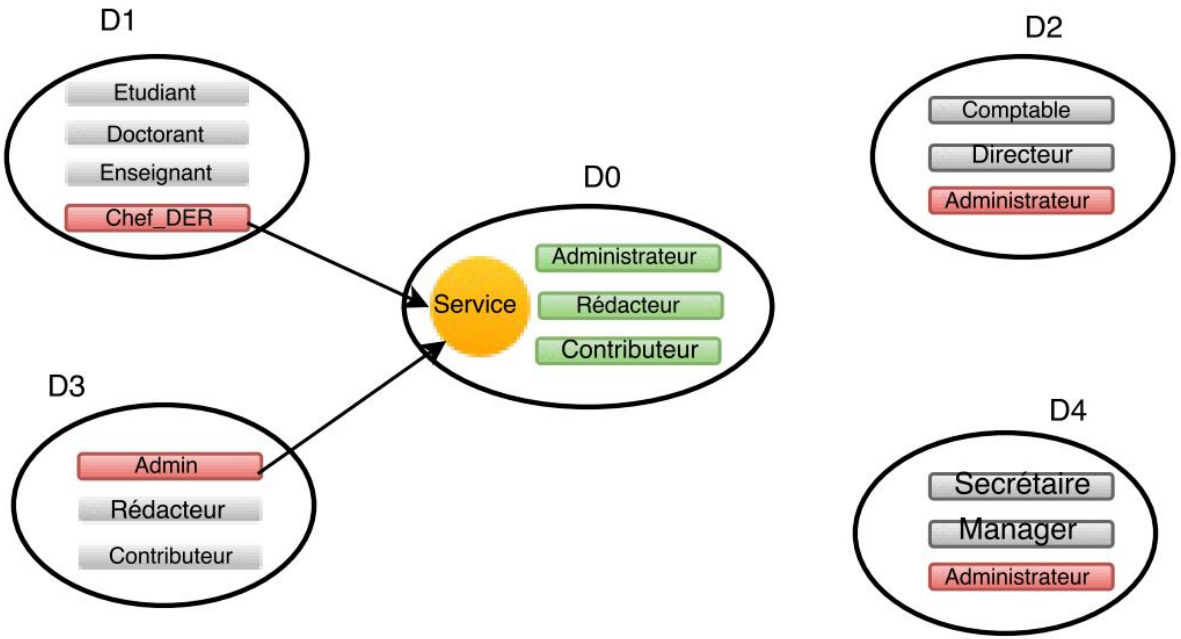
Comment accorder à la volée des droits d'accès à un utilisateur distant sur mes services ?

Contributions

Limitations : autorisation des utilisateurs externes

Exemple 1 : **Accès à un service**

les utilisateurs de D1,..,D4 doivent être autorisés à la volée sur le Service de D0 selon leurs rôles



- Chaque domaine implémente des modèles d'autorisation différents.
- Un rôle n'a pas la même signification dans les domaines
- l'autorisation d'un utilisateur externe dépend de son rôle

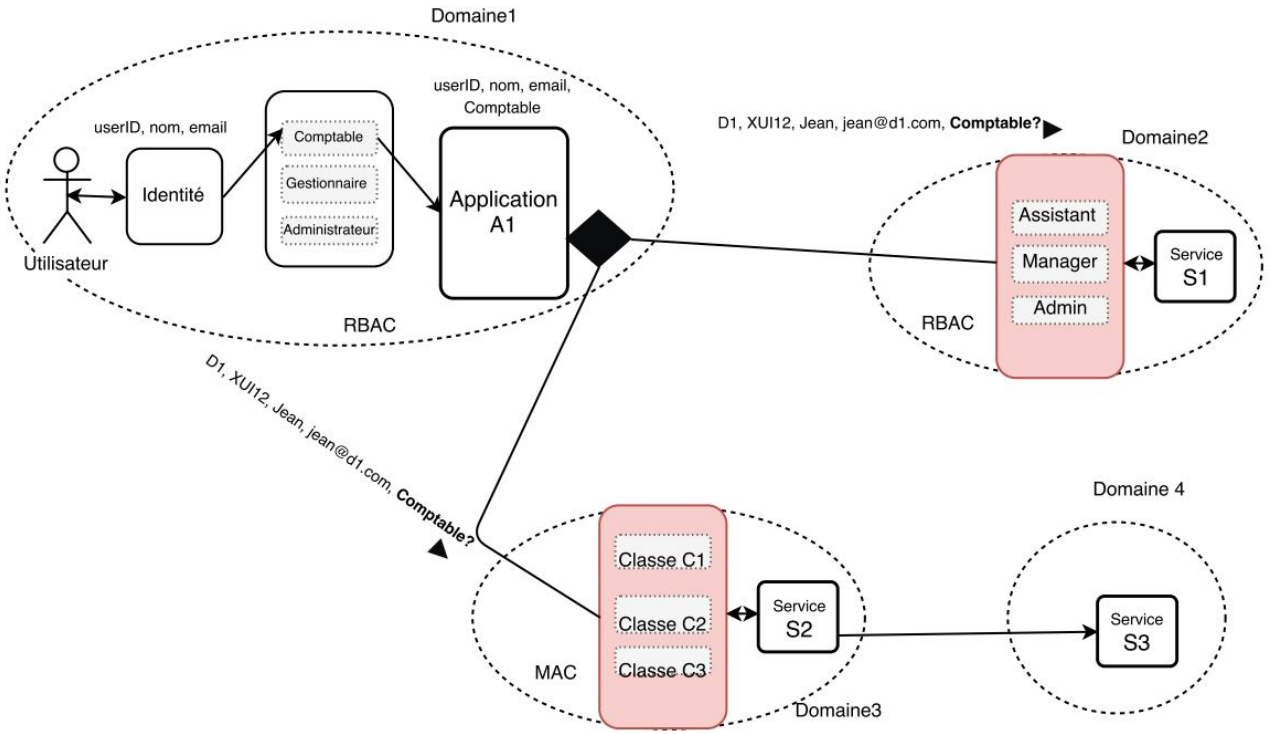
Besoin de correspondance cohérente entre rôles

Questions :

- A quel rôle de D0 correspond le rôle Chef-DER de D1 ?
- Que représente Chef-DER ?

Limitations : autorisation des utilisateurs externes

Exemple 2 : **Accès à plusieurs services** (et/ou en chaine)



Comment vérifier les autorisations de l'utilisateur *«D2, XUI12, Jean, ...»* sur les services S1, S2 et/ou S3 ?

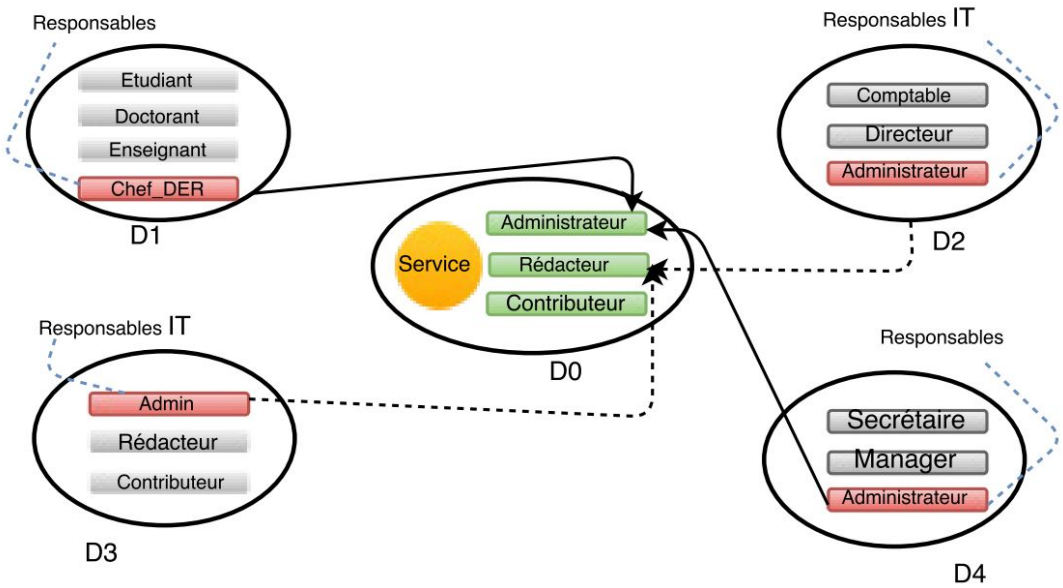
Besoin de correspondance cohérente entre rôles

Questions :

- A quoi correspond le rôle Comptable dans D2 , D3 et D4 ?
- Que représente un Comptable de D1 ?

Limitations : autorisation des utilisateurs externes

Correspondance des attributs d'autorisations



P2P Mapping

D1.Chef_DER = Administrateur

D4.Administrateur = Administrateur

D2.Administrateur = Rédacteur

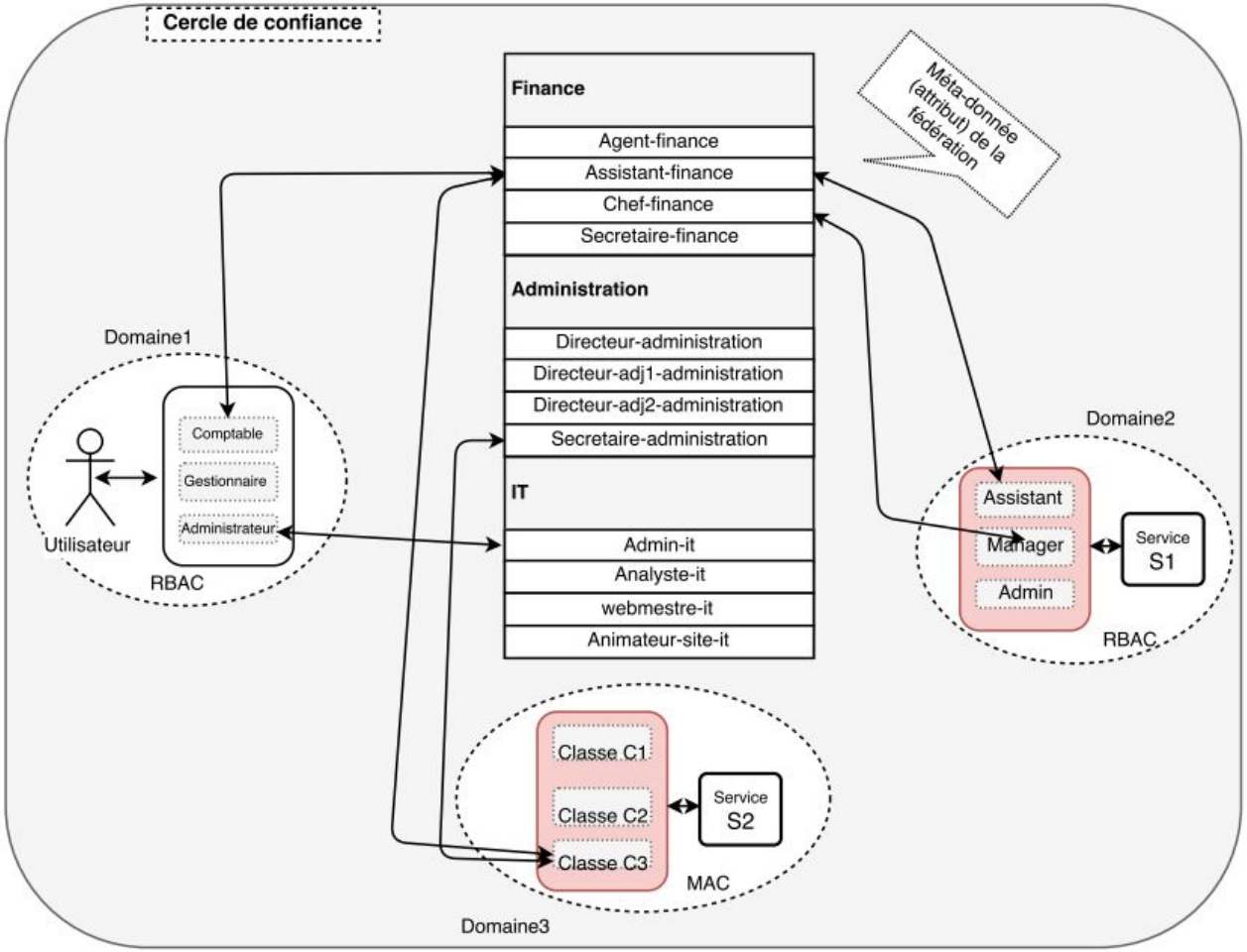
D3.Admin = Rédacteur

Problèmes

1. Pas de compréhension commune des attributs d'autorisation
2. L'octroi des autorisations difficile (sans la création d'une identité locale correspondante)
3. La correspondance point-à-point (P2P) est complexe et non évolutive

Vers une approche fédérée de correspondance

Proposition : Correspondance d'autorisations avec l'attribut fédéré

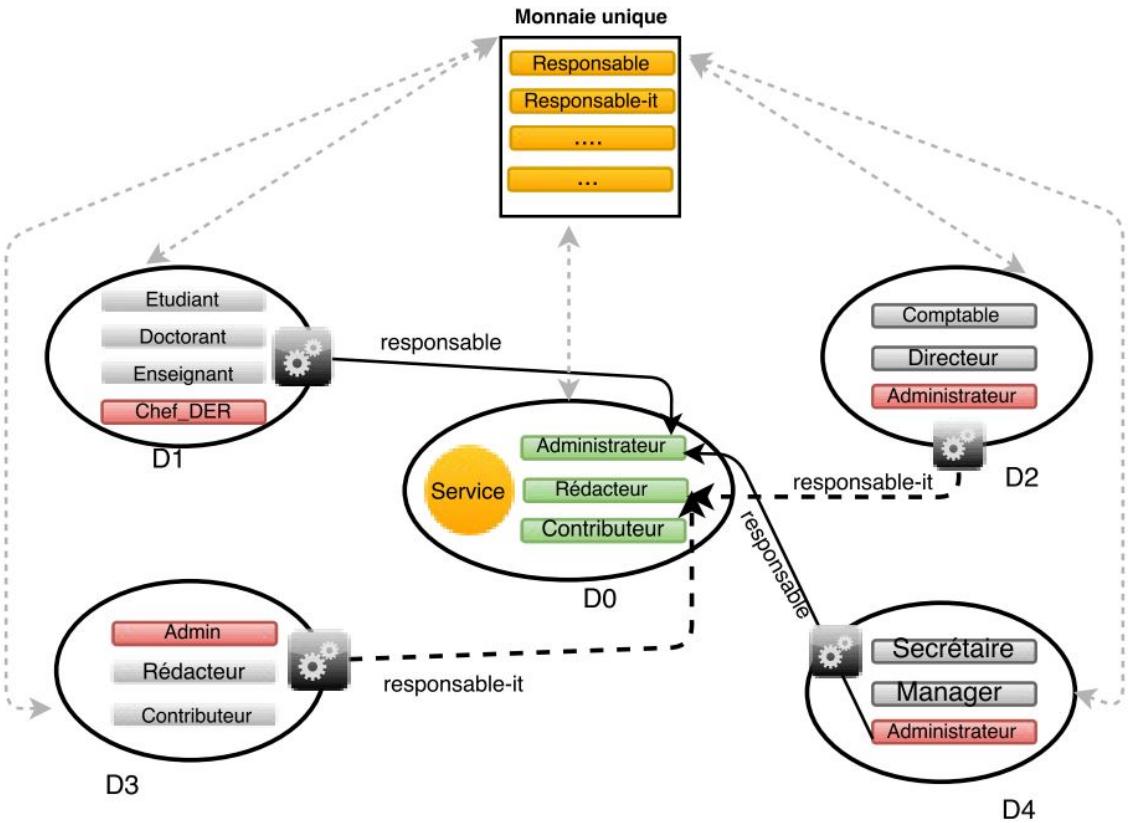


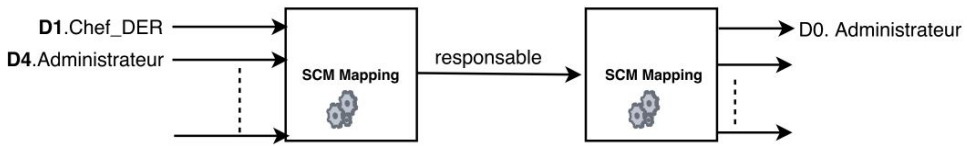
Attribut fédéré (*monnaie unique*)

- pour la correspondance entre les autorisations des domaines
- établi et convenu par tous les domaines
- disponible dans tous les domaines
- utilisé pour toutes interactions inter-domaines

Proposition : Correspondance d'autorisations avec l'attribut fédéré

Single Currency Mapping (SCM)





Avantages

1. Attributs et valeurs d'autorisation indépendants
2. Abstraction des modèles d'autorisation sous-jacents
3. Correspondance évolutive
4. Préserve la confidentialité d'informations de sécurité (rôles interne)
5. Composition de services sécurisés inter-domaines facilitée

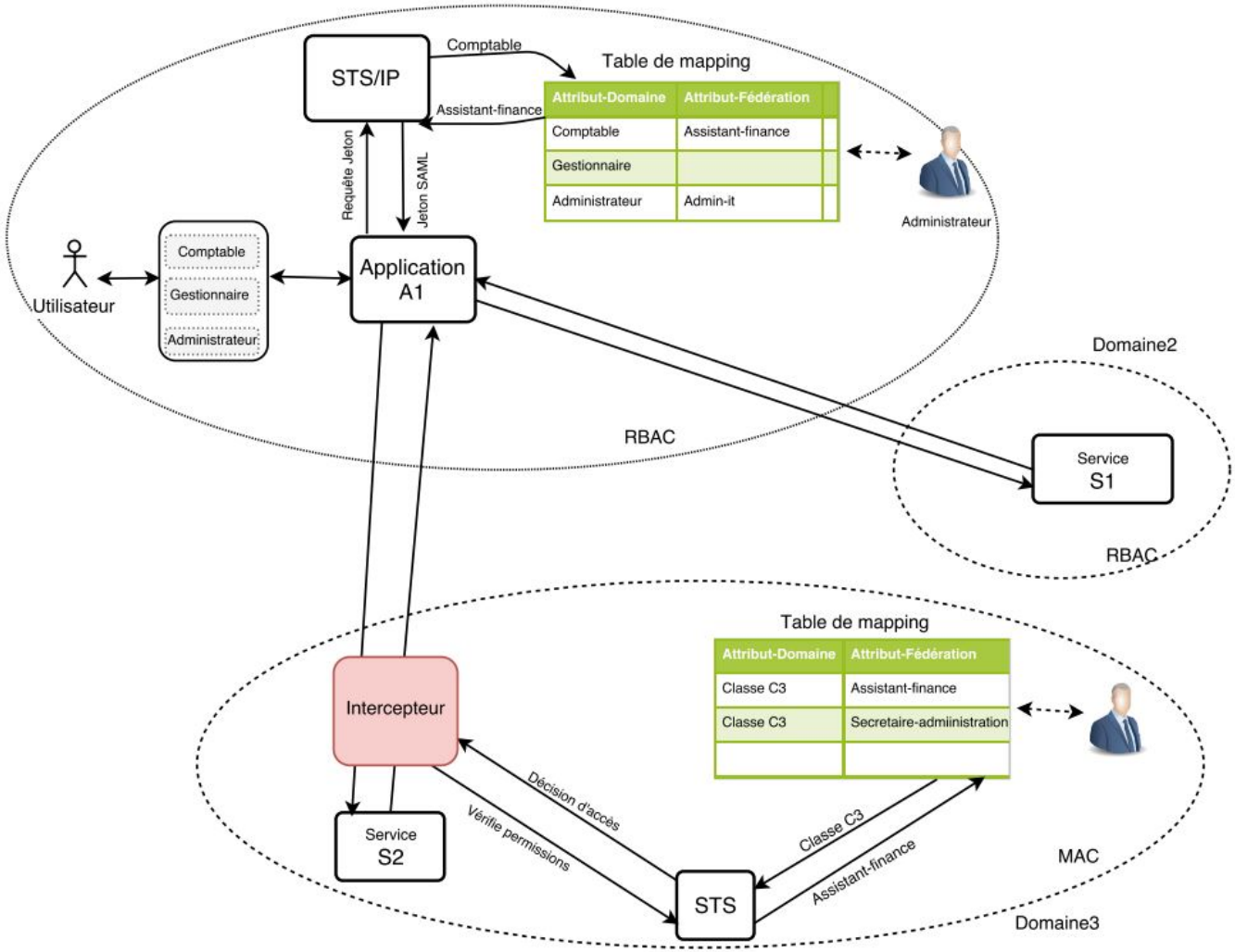
SCM Mapping (D0)

responsable = Administrateur

responsable-it = Rédacteur

Proposition : Correspondance d'autorisations avec l'attribut fédéré

Détails : Application de notre approche pour le contrôle d'accès aux services web

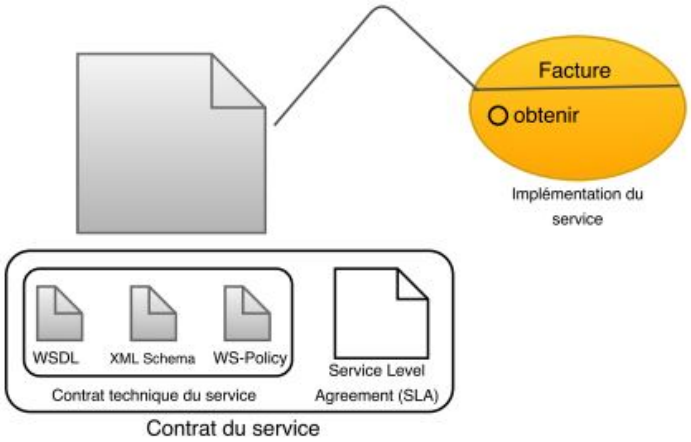


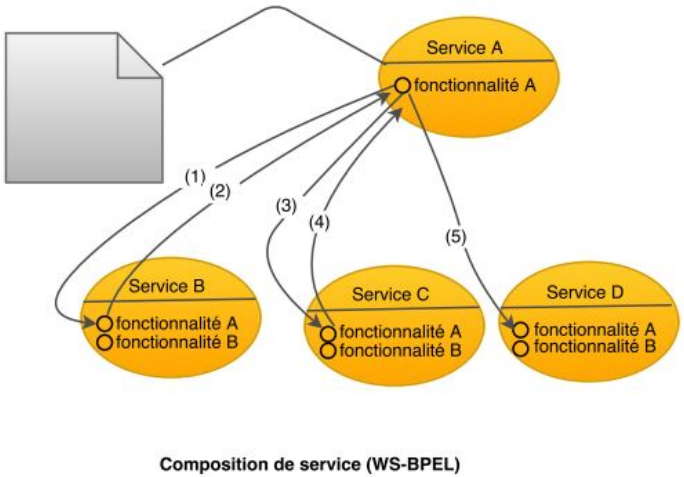
Expérimentation

Expérimentation

Service web ws-*

est un module logiciel auto-descriptif et autonome disponible via un réseau, tel qu'Internet, qui exécute des tâches, résout des problèmes ou effectue des transactions pour le compte d'un utilisateur ou d'une application [2].





Composition de service (WS-BPEL)

Sécurisation et invocation de services web sécurisés

- Sécurisation des messages échangés
- Mise en place de la confiance intra-domaine
- Contrôle d'accès (authentification, autorisation) intra-domaine

Composition de services web sécurisés inter-domaines

- Sécurisation des messages
- Mise en place de la confiance inter-domaines
- Contrôle d'accès inter-domaines
- Utilisation de (différents modèles autorisation)

Expérimentation

Spécifications de sécurité des services web

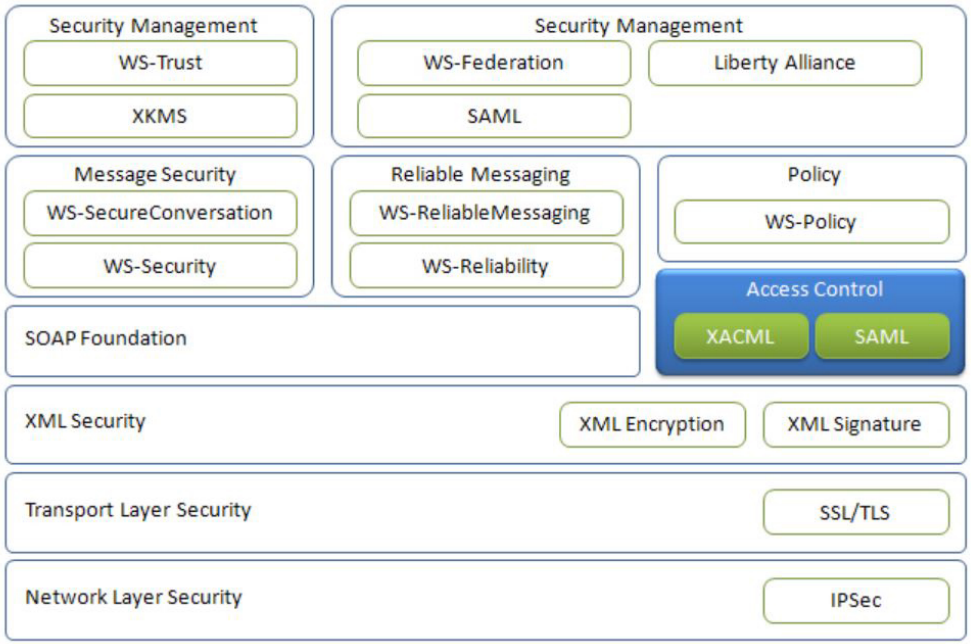


FIGURE – La pile de sécurité des services Web [3]

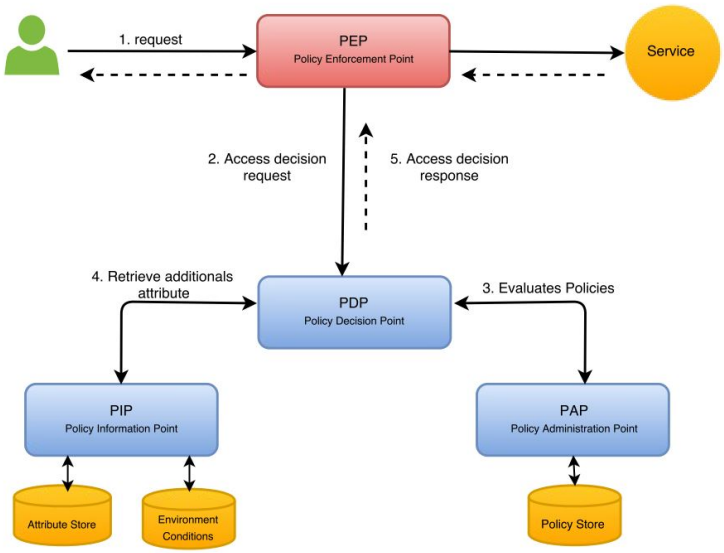
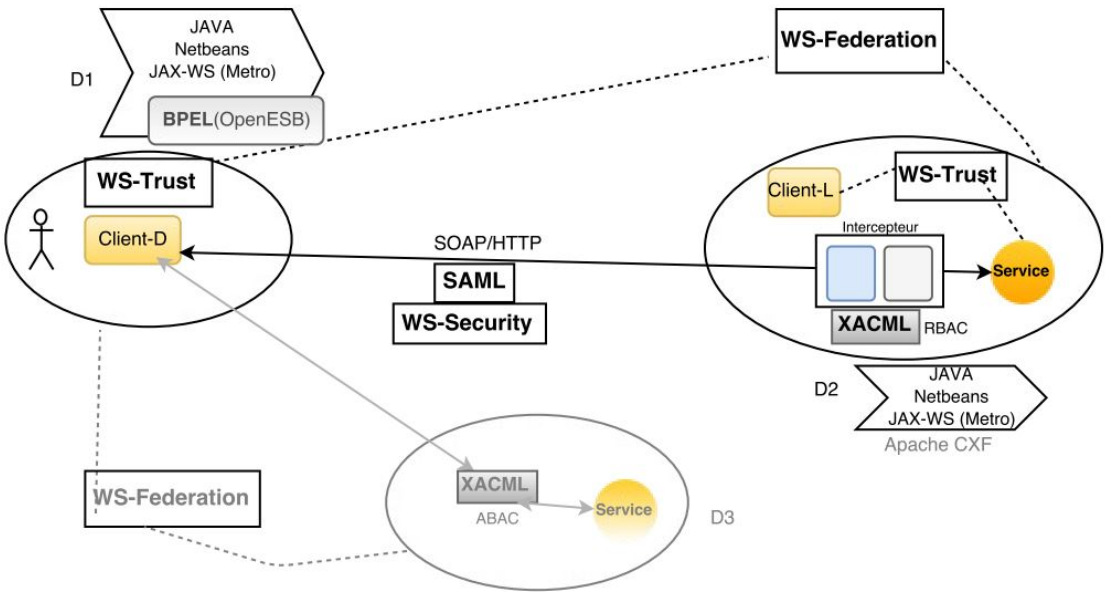


FIGURE – Architecture de contrôle d'accès (XACML)

Expérimentation

Outils et spécifications de sécurité utilisés



Outils utilisés

IDE/serveurs : **NetBeans**/Glassfish, Tomcat

Framework de développement de service :

- Java API for XML-Based Web Services (**JAX-WS RI**, *Apache CXF, Axis2*)

Framework de composition de service :

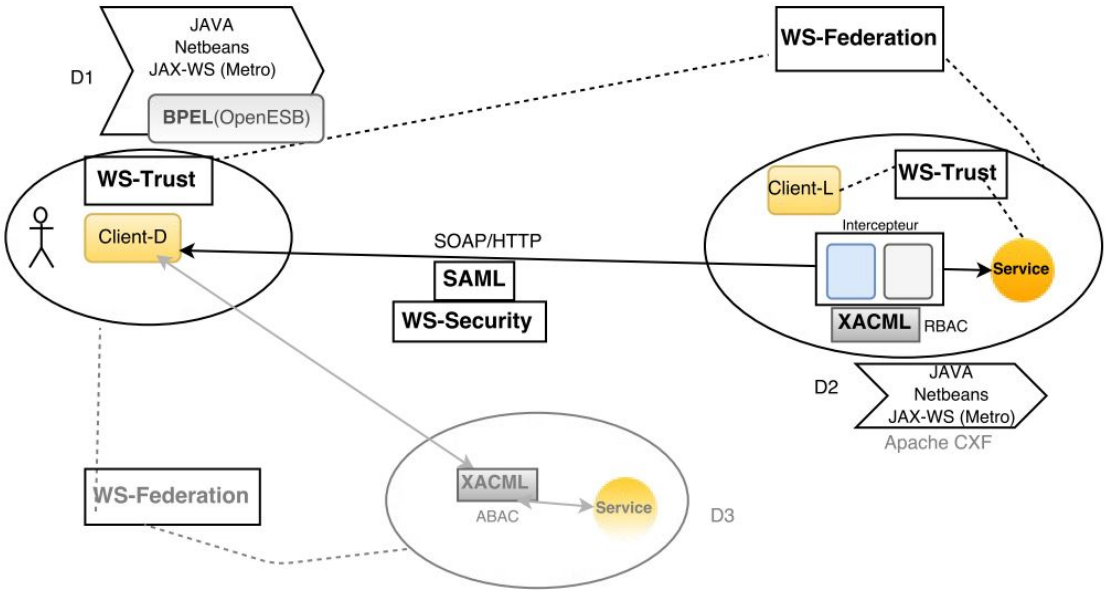
- **OpenESB** : (XML, WSDL, BPEL, Application composite etc...)

Framework de contrôle d'accès

- **OpenAM** : (gestion et fédération des identités, authentification (LDAP, X509), contrôle d'accès(XACML))

Expérimentation

Outils et spécifications de sécurité utilisés



Difficultés rencontrées :

1. Manque de documentations pratiques
   - contrôle d'accès
   - Configuration de la confiance intra et inter-domaines
   - composition de services web sécurisés
2. Configuration des framework de contrôle d'accès complexe et mal-documentée

Conclusion

Conclusion

Bilan

- Le contrôle d'accès aux services web inter-domaines s'effectue par la correspondance des attributs autorisations (ex : le rôle) des différents domaines
- le support d'attributs particuliers dans les domaines empêche une correspondance cohérente entre les attributs de ceux-ci.
- les solutions de correspondance point-à-point (peer-to-peer) sont complexes, coûteuses et non évolutives
- Nous proposons une approche fédérée, faiblement couplée et évolutive en utilisant des attributs fédérés (monnaie unique) indépendants pour établir les correspondances entre les autorisations des domaines
- Nous expérimentons notre approche avec les services web ws-* en JAVA sous NetBeans avec le framework JAX-WS en mettant en œuvre les spécifications de sécurité telles que WS-Sécurity, WS-Trust, WS-Federation, SAML, XACML.

Perspectives

- Composition des services sécurisés JAVA et .NET
- Composition d'une chaine de services web sécurisés

Références

David Chen and Nicolas Daclin.
Framework for enterprise interoperability.
In *Proc. of IFAC Workshop EI2N*, pages 77–88, 2006.

Michael P. Papazoglou.
Web services : principles and technology.
Pearson/Prentice Hall, Harlow, 2008.
OCLC : 255863191.

Anoop Singhal, Theodore Winograd, and Karen Scarfone.
Guide to secure web services.
NIST Special Publication, 800(95) :4, 2007.