Le service d’annuaire LDAP

M.BOUABID, 05-2015

Le concept d'annuaire (1/4)

- Un annuaire est comme une base de données.
  - on peut y mettre des information et les consulter
- Cependant un annuaire est spécialisé :
  - Dédié à la lecture plus qu'à l'écriture
  - L'accès aux données se fait par des recherches multicritères.
- Son objectif est de maintenir de façon cohérente et contrôlée une grande quantité de données.

Le concept d'annuaire (2/4)

- Exemples d'annuaire :
  - carnet d'adresses
  - annuaire téléphonique
    - regroupe différentes entrées contenant chacune des informations particulières : nom, prénom, numéro de téléphone et adresse.
    - Ces informations sont classées par ville, puis par code postale, puis enfin par nom.

Le concept d'annuaire (3/4)

- Voici les caractéristiques communes aux annuaires :
  - Un annuaire présente un **ensemble défini de données** (annuaire : nom, prénom, numéro de téléphone, adresse)
  - Il **organise** ces données (classées par département, villes, nom)
  - Il offre un service de **consultation**
  - Il peut **protéger** les données (liste rouge)
  - Il est **plus consulté** que mis à jour
  - Il est **disponible** de manière permanente

Le concept d’annuaire (4/4)

- Différences annuaires/SGBD
  - Sur un annuaire, les écritures sont plus rares que les lectures, ce qui n'est pas forcément le cas pour un SGBD
  - Un annuaire fournit une méthode de consultation standardisée, ce qui n'est pas le cas d'un SGBD.
  - Un annuaire LDAP organisent les données de manière arborescente, tandis que les bases de données le font au sein de tableaux à deux dimensions

L’annuaire LDAP

- **LDAP** : **L**ightweight **D**irectory **A**ccess **P**rotocol
- Hérite de l’annuaire X500 (proposé par l’ISO)
  - standard conçu par les opérateurs télécom pour interconnecter leurs annuaires téléphoniques
  - X500 adapté à l’internet
- LDAP, standard d’annuaire proposé en 1995, Actuellement Il s’agit de LDAP version 3

Quelques annuaires LDAP

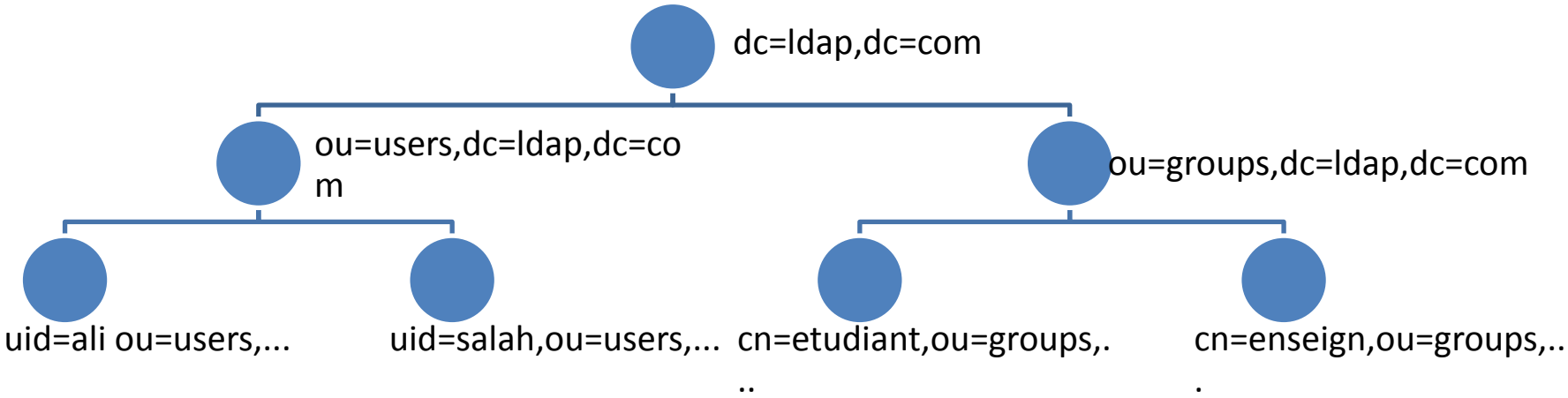
- Serveur Ldap:
  - **OpenLDAP** : **http://www.openldap.org**
  - **Apache Directory Server http://directory.apache.org**
  - **Sun (One/Java) Directory Server http://www.sun.com**
  - **Active Directory** : **http://www.microsoft.com**
- Clients LDAP
  - **phpLDAPadmin** : un client Web multiplateforme développé en PHP permettant de gérer facilement son annuaire LDAP
  - **Jxplorer** , **LDAPBrowser**: clients développés sous Java et donc indépendants du système d'exploitation
  - **LDAP Admin** : client pour windows

Concepts du protocole LDAP

- LDAP définit :
  - **Le protocole** : accéder à l'information contenue dans l'annuaire (serveur LDAP agit en tant qu'intermédiaire entre une source de données et un client.)
  - **Le modèle de nommage** : définit comment l'information est stockée et organisée
  - **Le modèle fonctionnel** : définit les services fournis par l'annuaire (recherche, ajout, ...)
  - **Le modèle d'information** : définit le type d'informations stockées
  - **Le modèle de sécurité** : définit les droits d'accès aux ressources

Organisation des données (modèle de nommage) 1/4

- Une représentation hiérarchique des données
- Exemple d'arborescence LDAP



Organisation des données (modèle de nommage) 2/4

- un élément marque son appartenance à l'élément supérieur en reprenant le nom, qu'il complète par le sien
- **"cn=etudiants,ou=groups,dc=ldap,dc=com"**

est situé sous l'élément "**ou=groups**" qui lui-même est situé sous l'élément "**dc=ldap,dc=com**".

- Chaque élément est appelé une entrée (an entry). Une entrée peut être un branchement (un noeud) ou un élément terminal (une feuille)

Termes à connaître

- Chaque élément possède un **DN** (Distinguished Name). Le DN est le nom complet de l'élément qui permet de le positionner dans l'arborescence. Il est unique dans l'annuaire.
  - Exemple : "cn=etudiants,ou=groups,dc=ldap,dc=com "
- Chaque élément possède également un **RDN** (Relative Distinguished Name). Le RDN est la partie du **DN** de l'élément qui est relative au **DN** supérieur. Le RDN d'un élément ne permet pas de l'identifier de manière absolue dans l'annuaire.
  - Exemple : "cn=etudiants"
- La **racine** est l'élément supérieur de tous les autres, c'est la base de l'arborescence.
  - Exemple : "dc=ldap,dc=com"

Organisation des données (modèle de nommage) 3/4

- Les DN de chaque entrées sont composés au moins d'un attribut de l'élément (par exemple "cn" [Common Name] ou "uid"[User IDentifier]) et de sa valeur. Un attribut est l'une des caractéristiques de cet élément.

- la racine choisie ici est composée du nom du domaine où est hébergé notre serveur LDAP, ldap.com, décomposé en "dc" (Domain Components) pour obtenir dc=ldap,dc=com.

Organisation des données (modèle de nommage) 4/4

- L'arbre se découpe ensuite en deux "ou" (Organisational Units) qui constituent deux branchements : "users" et "groups", dans lesquels nous trouvons ensuite les entrées feuilles de notre arbre : les utilisateurs et les groupes.

- Chacune des entrées de notre arbre correspond à un type de donnée particulier, défini par une classe d'objet

Accéder à l'annuaire (modèle fonctionnel) 1/4

- Il existe plusieurs types d'opérations que l'on peut effectuer sur l'annuaire
  - Rechercher une entrée suivant certains critères
  - S'authentifier
  - Ajouter une entrée
  - Supprimer une entrée
  - (Modifier une entrée)
  - Renommer une entrée

Accéder à l'annuaire (modèle fonctionnel) 2/4

- La **base** est le DN à partir duquel nous allons agir.
  - Pour une recherche, il s'agit du nœud à partir duquel est effectuée la recherche.
- La **portée** (scope) est le nombre de niveaux sur lesquels l'action va être effectuée. Il existe 3 niveaux différents :
  - **SUB** : l'action est effectuée récursivement à partir de la base spécifiée sur la totalité de l'arborescence.
  - **ONE** : l'action est effectuée sur un seul niveau inférieur par rapport à la base spécifiée (les fils directs). Si l'on effectuait une recherche avec la portée ONE à partir de
  - **BASE** : l'action est effectuée uniquement sur la base spécifiée..

Accéder à l'annuaire (modèle fonctionnel) 3/4

- Un **filtre** va permettre d'effectuer des tests de correspondance lors d'une recherche. Il s'agit en quelques sortes du critère de la recherche.
- Il existe 4 tests basiques, qui peuvent ensuite être combinés :
  - Le test d'égalité : **X=Y**
  - Le test d'infériorité : **X<=Y**
  - Le test de supériorité : **X>=Y**
  - Etc**.**

Accéder à l'annuaire (modèle fonctionnel) 4/4

- Les URLs LDAP
  - Interroger un annuaire LDAP à travers l'url suivante :
    - ldap[s]://serveur[:port][/[base[?[attributs à afficher][?[portée][?[filtre][?extensions]]]]]]
    - Exemple :

    ldap://localhost:389/ou=users,dc=martymac,dc=com?uid?sub

Les données contenues dans l'annuaire (modèle d'information) 1/4

- Un une entrée. Une entrée peut bien entendu contenir plusieurs attributs
- Exemple :entrée LDAP complète d'un compte utilisateur POSIX

```
dn: uid=ali,ou=users,dc=ldap,dc=com
objectClass: account
objectClass: posixAccount
cn: ali
uid: ali
uidNumber: 10001
gidNumber: 10001
homeDirectory: /home/ali
userPassword:: e0NSWVBUfWJjT29IUk5SbG1HbC4=
loginShell: /bin/sh
gecos: ali
```

Les données contenues dans l'annuaire (modèle d'information) 2/4

- L'attribut "**ou**" constitue une "**Organisational Unit**", c'est à dire une unité organisationnelle : en quelque sorte un regroupement.
  - Nous avons choisi d'en créer deux dans notre exemple : "users", et "groups "
- **L'objectClass** définit un regroupement d'attributs **obligatoires** ou **autorisés** pour une entrées
- Comment savoir quels sont les objectClass disponibles et quels attributs ils contiennent ?

→ la syntaxe et la liste des attributs connus de l'annuaire sont écrits dans ce que l'on appelle les "**schémas**".

Les données contenues dans l'annuaire (modèle d'information) 3/4

– l'objectClass posixAccount est défini dans le fichier **nis.schema**. Etudions une partie de ce fichier, livré avec OpenLDAP et situé dans **/etc/ldap/schema**

```
# [...]
attributetype ( 1.3.6.1.1.1.1.0 NAME 'uidNumber'
DESC 'An integer uniquely identifying a user in a domain'
EQUALITY integerMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 SINGLE-VALUE )
# [...]
objectclass ( 1.3.6.1.1.1.2.0 NAME 'posixAccount' SUP top AUXILIARY
DESC 'Abstraction of an account with POSIX attributes'
MUST ( cn $ uid $ uidNumber $ gidNumber $ homeDirectory )
MAY ( userPassword $ loginShell $ gecos $ description ) )
```

Le format LDIF

- Les données contenues dans l'annuaire sont présentées dans le format **LDIF** (**L**DAP **D**ata **I**nterchange **F**ormat)

- toute interaction avec un annuaire se fait par le biais de ce format : l'ajout, la modification, la suppression d'entrées, l'interrogation de l'annuaire y compris

Les données contenues dans l'annuaire (modèle d'information) 4/4

Exemple :

```
# [...]
dn: cn=etudiants,ou=groups,dc=ldap,dc=com
objectClass: posixGroup
cn: etudiants
gidNumber: 10001

dn: uid=ali,ou=users,dc=ldap,dc=com
objectClass: account
objectClass: posixAccount
cn: ali
uid: ali
uidNumber: 10001
gidNumber: 10001
homeDirectory: /home/ali
userPassword:: e0NSWVBUfWJjT29IUk5SbG1HbC4=
loginShell: /bin/sh
gecos: ali
description: ali
# [...]
```

La sécurité (modèle de sécurité) 1/3

- L'authentification simple, le binding
  - opérations préalables à l'interrogation de l'annuaire
  - Le client envoie alors le DN d'un compte contenu dans l'annuaire lui-même, ainsi que le mot de passe associé.
  - On pourra par la suite appliquer des droits particuliers sur ce compte en utilisant les ACLs
  - Exemple : la fonctionnalité de liste rouge pour un annuaire téléphonique

La sécurité (modèle de sécurité) 2/3

- Les ACLs (Access Control Lists)
  - interviennent après la notion de binding.
  - Il sera possible de donner des droits de lecture, d'écriture (ou d'autres droits divers) sur des branches particulières de l'annuaire au compte connecté.

La sécurité (modèle de sécurité) 3/3

- Le chiffrement des communications (SSL/TLS)
  - chiffrer le canal de communication entre l'application cliente et l'annuaire via SSL (Secure Socket Layer, ou TLS - Transport Layer Security)
  - garantir la confidentialité des données et éviter qu'un tiers n'écoute les communications sur le réseau