

R304

Gestion d'annuaires

Sami Evangelista
IUT de Villetaneuse
Département Réseaux et Télécommunications
2025–2026

<http://www.lipn.univ-paris13.fr/~evangelista/cours/R304>

Sources des images utilisées dans ce document:

<http://www-lipn.univ-paris13.fr/~evangelista/cours/credits.html>

Ce document est mis à disposition selon les termes de la licence Creative Commons "Attribution – Pas d'utilisation commerciale – Partage dans les mêmes conditions 3.0 non transposé".



- ▶ Cours de Marisol Rodriguez Perez et Xavier Monnin
<https://lipn.fr/~monnin/Enseignement/ServicesAnnuaires/>
- ▶ Tutorial LDAP de Laurent Mirtain
<https://www-sop.inria.fr/members/Laurent.Mirtain>
- ▶ OpenLDAP — Implantation open source de LDAP
<https://www.openldap.org>

Plan

3/21

1. Introduction

2. LDAP

Caractéristiques d'un annuaire

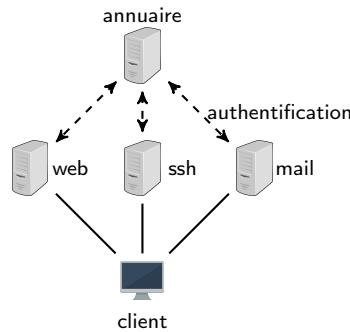
4/21

annuaire = base de données

- ▶ avec beaucoup de lectures, peu de modifications
- ▶ distribuée (⇒ répartie sur plusieurs serveurs)
- ▶ répliquée (⇒ recopiée sur plusieurs serveurs)
- ▶ consultée par d'autres services : web, mail, ... pour :
 - ▶ l'authentification
 - ▶ la consultation des données des utilisateurs (nom, adresse, ...)

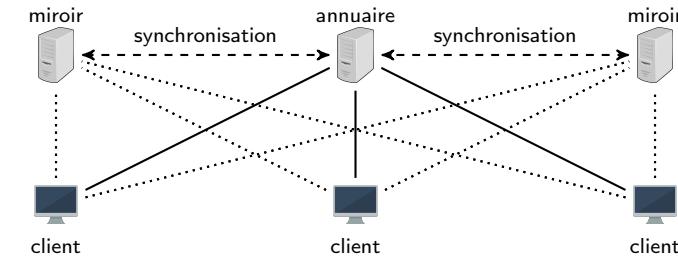
...

exemple : authentification des clients par les serveurs via l'annuaire



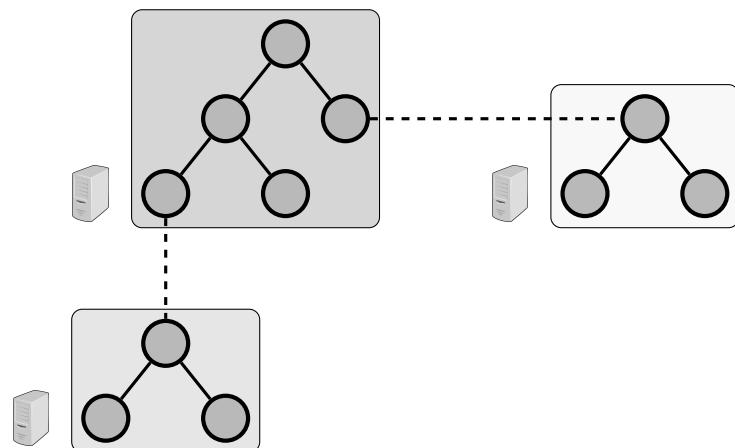
► annuaire = nœud central

- ⇒ en cas de panne, les autres services ne répondent plus
- ⇒ besoin de mécanismes de tolérance aux pannes



- mécanisme de synchronisation (i.e., réPLICATION) périodique entre l'annuaire et ses miroirs
- les clients consultent un miroir en cas de panne du serveur principal

exemple : un annuaire arborescent réparti sur 3 serveurs



- mécanisme de délégation entre serveurs (ex : DNS)
- facilite la gestion de grandes bases de données
- allège les serveurs

- spécialisés
 - DNS — Domain Name System
association nom ↔ adresse IP
 - NIS — Network Information Service
centralisation des noms d'hôtes (/etc/hosts), des comptes utilisateurs (/etc/passwd), des groupes (/etc/group), ..., d'un réseau
- généralistes
 - X.500
 - LDAP

- ▶ norme de l'UIT-T
- ▶ approuvée en 1988
- ▶ définition d'un service d'annuaire
 - ▶ normalisé pour faciliter l'échange de données
 - ▶ extensible pour pouvoir représenter tout type de données
- ▶ repose sur le modèle OSI

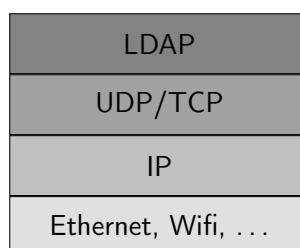
1. Introduction

2. LDAP

Présentation

11/21

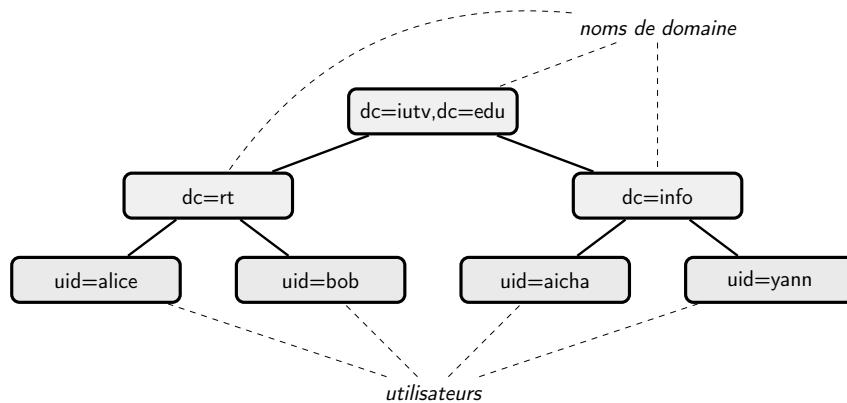
- ▶ Lightweight Directory Access Protocol
- ▶ création en 1993 (université du Michigan)
- ▶ normalisé ensuite par l'IETF
- ▶ version actuelle : v3 (RFC 2251, décembre 1997)
- ▶ au début : simplification de DAP (protocole d'accès à des annuaires X.500)
- ▶ maintenant : système complet avec protocoles et annuaires
- ▶ repose sur le modèle TCP/IP
- ▶ port UDP/TCP par défaut : 389 (636 si on utilise TLS)



Structuration des annuaires

12/21

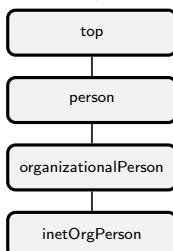
- ▶ Un annuaire LDAP a une structure **arborescente**.
- ▶ Chaque nœud (ou **objet**) de l'arbre représente une entrée de l'annuaire.
- ▶ Un objet peut être :
 - ▶ un objet abstrait (un domaine DNS, un mot de passe, ...);
 - ▶ ou un objet du monde réel (une personne, un ordinateur, ...).
- ▶ Un objet
 - ▶ est identifié dans l'arbre par son **DN** (Distinguished Name);
 - ▶ a divers **attributs** (nom, adresse IP, numéro de téléphone, ...);
 - ▶ et peut être d'une (ou de plusieurs) **classe(s)**.
- ▶ Un attribut peut être multi-valué : plusieurs valeurs peuvent lui être affectées (p.ex. : plusieurs adresses IP pour un hôte).



- ▶ 4 utilisateurs répartis sur 2 domaines
 - rt.iutv.edu
 - info.iutv.edu
 - ▶ DN d'alice : uid=alice,dc=rt,dc=iutv,dc=edu

Modèle de classe

- ▶ Chaque classe d'objet définit des attributs :
 - ▶ obligatoires (i.e., des attributs que des objets de cette classe doivent avoir).
 - ▶ et facultatifs (resp. peuvent avoir).
 - ▶ modèle hiérarchique/arborescent :
 - ▶ une classe a au plus une super classe ;
 - ▶ et peut avoir plusieurs sous classes.
 - ▶ `top` est la classe de plus haut niveau.
 - ▶ L'attribut `objectClass` permet d'attribuer une (des) classe(s) à un objet.
 - ▶ Un objet d'une classe C
 - ▶ doit avoir tous les attr. obligatoires de C ou d'une des classes dont C hérite,
 - ▶ et peut avoir tous les attr. facultatifs de C ou des classes dont C hérite.



Un objet de la classe `inetOrgPerson` a tous les attributs des classes `inetOrgPerson`, `organizationalPerson`, `person` et `top`.

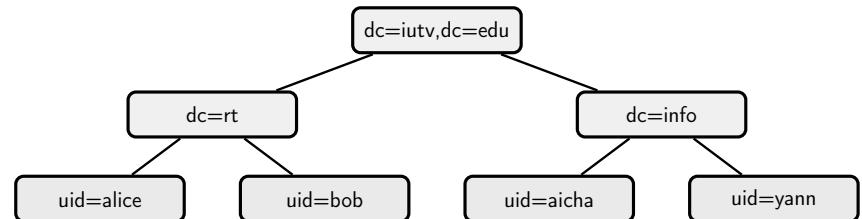
- ▶ cn = common name
 - ▶ dc = domain component
 - ▶ o = organization
 - ▶ ou = organizational unit
 - ▶ sn = surname
 - ▶ uid = user identifier

Quelques classes usuelles

Classe	Super classe		Attributs obligatoires	Attributs facultatifs
top	—		objectClass	—
dcObject	top		dc	—
organization	top		o	postalAddress description telephoneNumber ...
person	top	sn cn		description telephoneNumber ...
organizationalPerson	person	—		title postalAddress postalCode ...
inetOrgPerson	organizationalPerson	—		mail mobile homePhone ...

(Pour voir les définitions exactes : /etc/ldap/schema)

- ▶ LDIF = LDAP Data Interchange Format
 - ▶ Fichiers texte permettant de représenter
 - les objets d'un annuaire ;
 - et les opérations de modification sur un annuaire.
 - ▶ La définition d'un objet débute avec son DN (dn: ...).
 - ▶ On a ensuite les attributs de l'objet sous la forme :
- attr: value
- ▶ Une ligne vide marque la fin de la définition d'un objet.



```

dn: dc=iutv , dc=edu
objectClass: top
objectClass: dcObject
dc: iutv
...

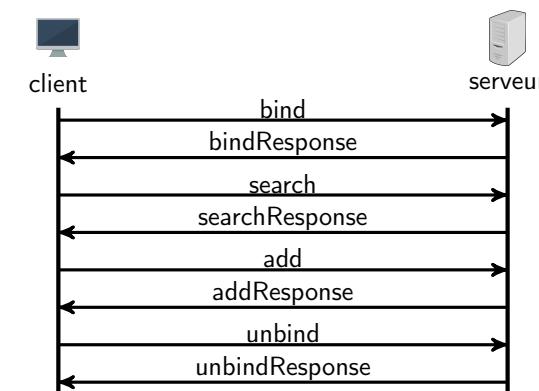
dn: uid=alice , dc=rt , dc=iutv , dc=edu
objectClass: top
objectClass: person
uid: alice
...
  
```

Opérations client ↔ serveur principales :

Opération	Description
search	recherche multi-critères
add	ajout d'un objet
delete	suppression d'un objet
modify	modification des attributs d'un objet
rename	modification du DN d'un objet
bind	connexion
unbind	déconnexion

Opérations serveur ↔ serveur principales :

Opération	Description
replication	duplication de l'annuaire (ou d'une partie)
referral	délégation d'une branche de l'annuaire



Possibilité d'envoyer plusieurs requêtes durant la même connexion LDAP.

Syntaxe	Signification	Exemple(s)
attr=valeur	—	o=IUT de Villetaneuse o=IUT*
attribut *= (!(c)) (&(c1)(c2)) ((c1)(c2))	attribut défini pour l'objet <u>non</u> c c1 <u>et</u> c2 c1 <u>ou</u> c2	name=* (!(o=IUT*)) (&(mail=*)(sn=*_paul*)) ((uid=alice)(uid=bob))

(Pour les opérateurs | et &, on peut avoir plus de 2 conditions.)