Gestion d'annuaires unifiés

OpenLDAP : Opérations LDAP et clients ligne de commande

Luca SAIU

http://ageinghacker.net

Département Réseaux et Télécommunications
IUT de Villetaneuse, Université Paris 13

Décembre 2018

Sommaire

Méta-syntaxe

- [*foo*]
  foo est optionnel ;
- Alternatives :
  - {*foo*, *bar*}
    foo ou bien *bar*.
  - {*foo*, *bar*, *quux*}
    foo ou bien *bar* ou bien *quux* ;
  - (banalement généralisée à quatre ou plus alternatives)
- *foo*…
  Un ou plusieurs *foo*.
- (dans mes slides) la couleur bleu indique une méta-variable.

Les caractères «[», «]», «{», «}» et «,» ne sont pas à écrire. Ils sont une façon d'indiquer une syntaxe.

Sur les RFCs décrivant LDAP la méta-syntaxe est précise, mais un peu plus lourde [Crocker (Ed.) and Overell, 2005].

Rappel : nous sommes sur Unix !

Il faut toujours tenir compte des expansions shell quand l'on écrit une ligne de commande.

- Les paramètres du shell doivent être donnés dans une seule chaîne de caractères, sans espaces ou retours à la ligne. Vous connaissez les *techniques d'échape* du shell («\», «'», «"», «<<EOF»).
  - Example :

```
$ ls ce fichier a des espaces dans son nom
ls: cannot access 'ce': No such file or directory
ls: cannot access 'fichier': No such file or directory
ls: cannot access 'a': No such file or directory
...
$ ls "ce fichier a des espaces dans son nom"
'ce fichier a des espaces dans son nom'
```

  - Certains outils de OpenLDAP acceptent des DNs dans la ligne de commande. Les DNs peuvent contenir des espaces ou d'autres caractères spéciaux.

L'outil `ldapadd`

Documentation complète sur le manuel de OpenLDAP et sur la page de `man`, comme d'habitude. Ce vaut pour tout outil.

```
ldapadd   [-h host:port]
          [-v]
          [-D bindname]
          [{-W,-w secret}]
          [-f ldifpathname]
```

Le *bindname* est le DN d'un utilisateur ; le rootdn spécifié dans la configuration ou, vous allez voir aujourd'hui, le DN d'une entrée ayant objectclass `person`.

Avec `-W` le mot de passe est demandé intéractivement.

Si l'option `-f` *ldifpathname* n'est pas donnée, `ldapadd` utilise le *standard input* au lieu d'un fichier. Le format est en tout cas LDIF.

L'outil ldapdelete

```
ldapdelete   [-h host:port]
             [-v]
             [-D bindname]
             [{-W,-w secret}]
             [DN...]
```

`-v` conseillé dans ce cas, pour avoir des retours.

Si les *DN*s ne sont pas donnés, accepte des DNs sur le standard input, un par ligne (donc, pas en LDIF dans ce cas).

L'outil ldapsearch

```
ldapsearch   [-h host:port]
             [-v]
             [-D bindname]
             [{-W,-w secret}]
             -b branche
             [filtre...]
```

Remarquez que la branche est obligatoire ici. Ce n'est pas le cas pour les autres outils que j'ai montré. Pourquoi ?

Si parfois vous êtes obligés de donner un filtre même si non nécessaire, vous pouvez utiliser des filtres banals comme « `objectclass=*` ».

L'outil ldapmodify

```
ldapmodify   [-h host:port]
             [-v]
             [-D bindname]
             [{-W,-w secret}]
             [-f ldifpathanme]
```

Une opération complexe. Le type d'opération est spécifié dans un attribut `changetype` dans l'entrée en LDIF.
`changetype` peut être `modify`, `delete`, `modrdn`. Selon le type d'opération, on spécifie des autres attributs avec les détails du changement demandé—encore, dans l'entrée en LDIF). Exemple (incomplet : *qu'est-ce que j'ai omis ?*) pour ajouter un attribut :

```
changetype: modify
add: telephonenumber
telephonenumber: (01) 113213342
```

Certains autres outils sont en réalité des façon simplifiées, et moins

L'outil ldapmodrdn

À découvrir par vous (très facile, à ce point).

Pourquoi ldapmodrdn ?

Demo

[Demo]

Bibliography I


Crocker (Ed.), D. and Overell, P. (2005). Augmented BNF for Syntax Specifications : ABNF. RFC 4234 (Draft Standard). Obsoleted by RFC 5234. (Une expliquation de la meta-syntaxe utilisée dans les autres RFCs. C'est officiellement remplacé par un autre RFC, mais il s'agit du document directement cité par les RFCs décrivant LDAP ; et c'est lisible).


Harrison (Ed.), R. (2006). Lightweight Directory Access Protocol (LDAP) : Authentication Methods and Security Mechanisms. RFC 4513 (Proposed Standard).


Legg (Ed.), S. (2006). Lightweight Directory Access Protocol (LDAP) : Syntaxes and Matching Rules. RFC 4517 (Proposed Standard).

Bibliography II

- Saiu, L. (2018). La page web de mes cours. http://ageinghacker.net/teaching
  La page web officielle du cours contient des pointeurs à des ressources web, et une copie des mes transparents.
- Sciberras (Ed.), A. (2006). Lightweight Directory Access Protocol (LDAP) : Schema for User Applications. RFC 4519 (Proposed Standard).
- Sermersheim (Ed.), J. (2006). Lightweight Directory Access Protocol (LDAP) : The Protocol. RFC 4511 (Proposed Standard).
- Smith (Ed.), M. and Howes, T. (2006a). Lightweight Directory Access Protocol (LDAP) : String Representation of Search Filters. RFC 4515 (Proposed Standard).

Bibliography III

- Smith (Ed.), M. and Howes, T. (2006b). Lightweight Directory Access Protocol (LDAP) : Uniform Resource Locator. RFC 4516 (Proposed Standard).
- Zeilenga, K. (2006). Lightweight Directory Access Protocol (LDAP) : Internationalized String Preparation. RFC 4518 (Proposed Standard).
- Zeilenga (Ed.), K. (2006a). Lightweight Directory Access Protocol (LDAP) : Directory Information Models. RFC 4512 (Proposed Standard).
- Zeilenga (Ed.), K. (2006b). Lightweight Directory Access Protocol (LDAP) : String Representation of Distinguished Names. RFC 4514 (Proposed Standard).

Bibliography IV


Zeilenga (Ed.), K. (2006c). Lightweight Directory Access Protocol (LDAP) : Technical Specification Road Map. RFC 4510 (Proposed Standard).