AGUILAR_Flavien_LDAP_TP

Flavien AGUILAR

01/12/2020

I - Installation et premier contact :

1) Nous allons commencer par installer les outils nécessaires :

```
root@debian:/home/test# apt install slapd ldap-utils
```

Pour installer apachedirectorystudio, je vais utiliser le gestionnaire de paquet yay car j'utilise une distribution archlinux sur mon ordinateur

```
yay -s apachedirectorystudio
```

2) Une fois les paquets installés, nous allons vérifier la présence du port 389 en attente de connexion :

```
root@debian:/home/test# ss -nat
State        Recv-Q       Send-Q             Local Address:Port
                 Peer Address:Port
LISTEN       0            128                      0.0.0.0:389
                        0.0.0.0:*
LISTEN       0            128                      0.0.0.0:22
                         0.0.0.0:*
ESTAB        0            0                  192.168.1.131:22
                   192.168.1.79:54598
LISTEN       0            128                         [::]:389
                          [::]:*
LISTEN       0            128                         [::]:22
                            [::]:*
```

Sur la première ligne, nous pouvons voir que le port 389 est en écoute.

Nous pouvons aussi tester la présence du processus slapd :

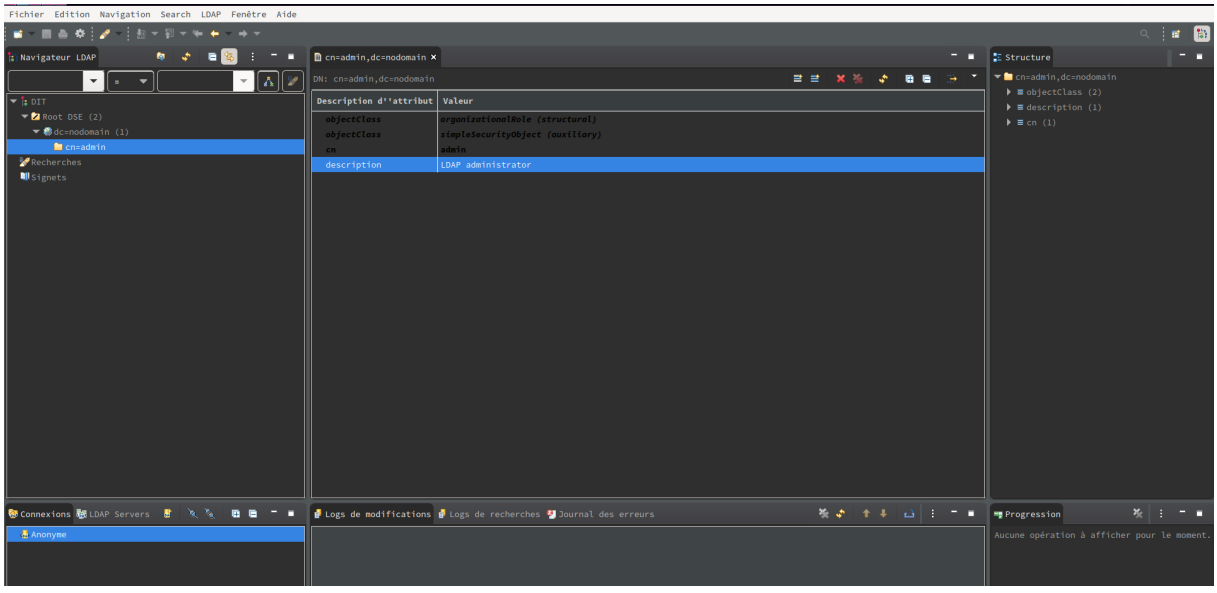
```
root@debian:/home/test# systemctl status slapd
 slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory
     Access Protocol)
   Loaded: loaded (/etc/init.d/slapd; generated)
   Active: active (running) since Tue 2020-12-01 09:38:08 CET; 11min
      ago
     Docs: man:systemd-sysv-generator(8)
    Tasks: 3 (limit: 1147)
   Memory: 3.3M
```

```
8     CGroup: /system.slice/slapd.service
9             5642 /usr/sbin/slapd -h ldap:/// ldapi:/// -g openldap -u
                 openldap -F /etc/ldap/slapd.d
10
11  déc. 01 09:38:08 debian systemd[1]: Starting LSB: OpenLDAP standalone
        server (Lightweight Directory Acc
12  déc. 01 09:38:08 debian slapd[5641]: @(#) $OpenLDAP: slapd  (Nov 17
        2020 01:23:45) $
13                                                  Debian OpenLDAP
                                                       Maintainers <pkg-
                                                       openldap-devel@lists.
                                                       ali
14  déc. 01 09:38:08 debian slapd[5642]: slapd starting
15  déc. 01 09:38:08 debian slapd[5636]: Starting OpenLDAP: slapd.
16  déc. 01 09:38:08 debian systemd[1]: Started LSB: OpenLDAP standalone
        server (Lightweight Directory Acce
```

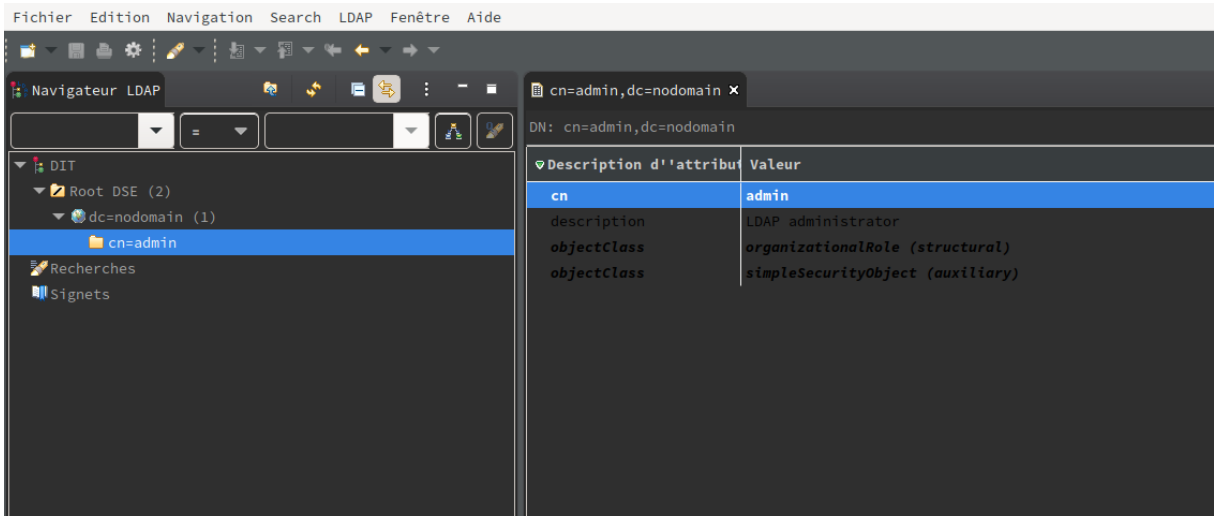
3) Nous allons chercher l'option olcSuffix :

```
1  root@debian:/home/test# cat /etc/ldap/slapd.d/cn\=config/olcDatabase
       \=\{1\}mdb.ldif | grep olcSuffix
2  olcSuffix: dc=nodomain
```

4) Nous allons ouvrir une connexion anonyme sur le serveur à l'aide d'apache directory studio :



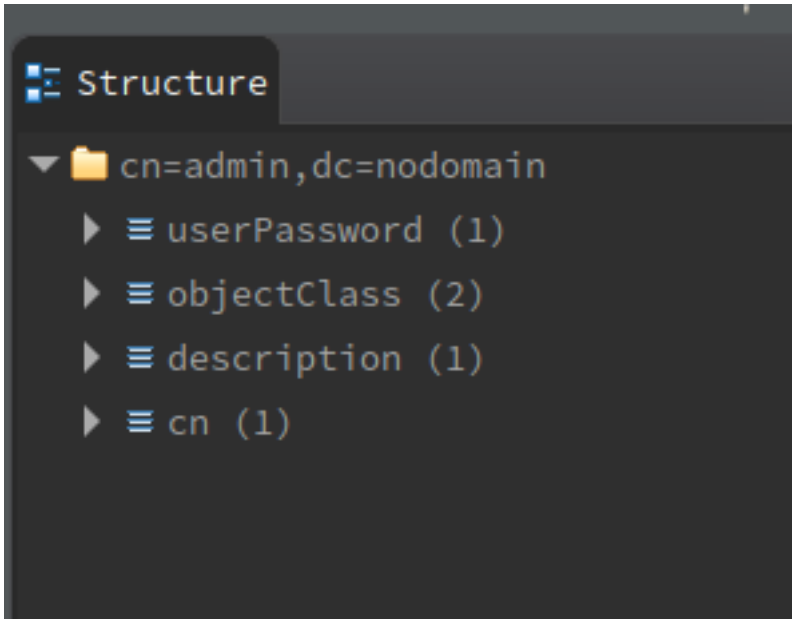
5) L'objet admin se trouve :



6) Je me reconnecte en admin :

```
login = cn=admin,dc=nodomain
password = root
```

7) Les attributs visibles sont :



8) Les olcAccess sont :

```
olcAccess: {0}to attrs=userPassword by self write by anonymous auth by
    * non
 e
olcAccess: {1}to attrs=shadowLastChange by self write by * read
olcAccess: {2}to * by * read
```

Les ACL permettent de fixer les droits sur l'annuaire, c'est pour cela qu'en admin, nous avons les droits d'écriture sur le champ userPassword, mais qu'en anonyme, nous ne pouvons même pas lire ce champ.

9) Nous allons créer le fichier ou-personnes.ldif :

```
root@debian:/home/test/tp-ldap# touch ou-personnes.ldif
```

Et nous allons y ajouter les éléments :

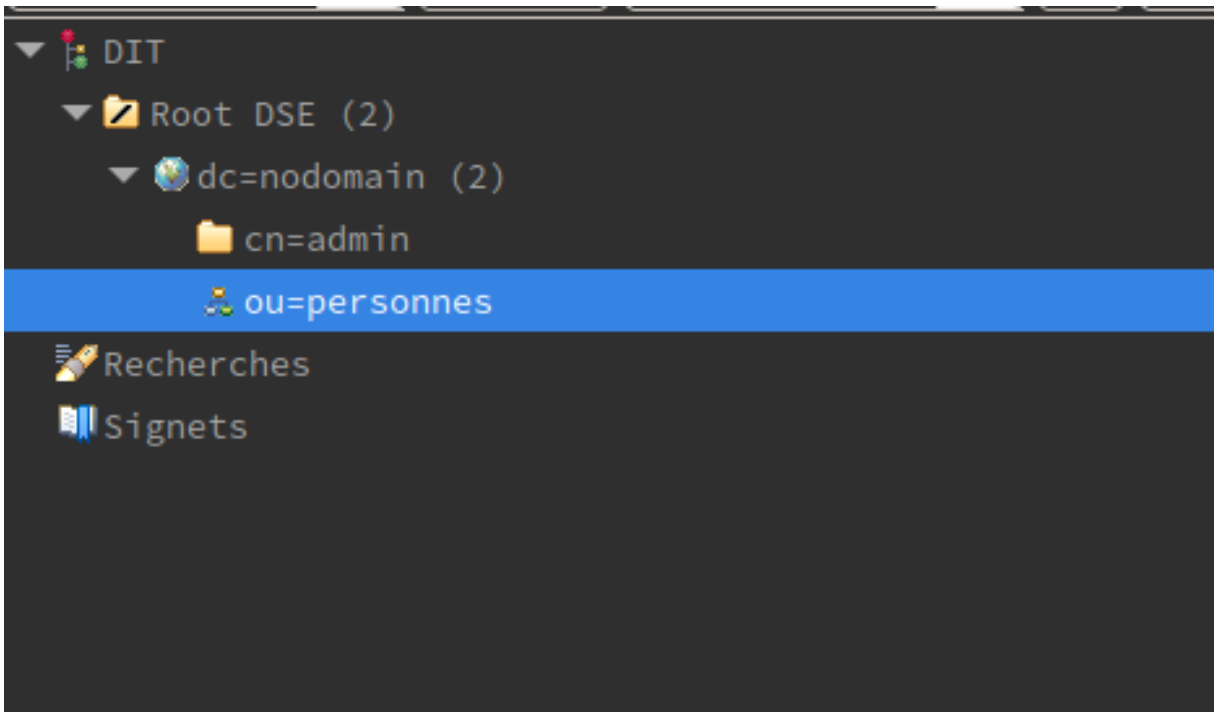
```
root@debian:/home/test/tp-ldap# cat ou-personnes.ldif
dn: ou=personnes,dc=nodomain
changetype: add
objectclass: top
objectclass: organizationalUnit
ou: personnes
description: Unite organisationnelle dediee aux personnes physiques
```

Nous pouvons maintenant faire le ldapmodify :

```
root@debian:/home/test/tp-ldap# ldapmodify -v -x -D cn=admin,dc=
    nodomain -W -h localhost -f ou-personnes.ldif
ldap_initialize( ldap://localhost )
Enter LDAP Password:
add objectclass:
    top
    organizationalUnit
add ou:
    personnes
add description:
    Unite organisationnelle dediee aux personnes physiques
adding new entry "ou=personnes,dc=nodomain"
modify complete
```

Le retour de la commande nous indique que les objets ont bien été ajoutés à l'annuaire.

Nous pouvons voir sur apachedirectorystudio que l'organizationalUnit personnes est rajouté :



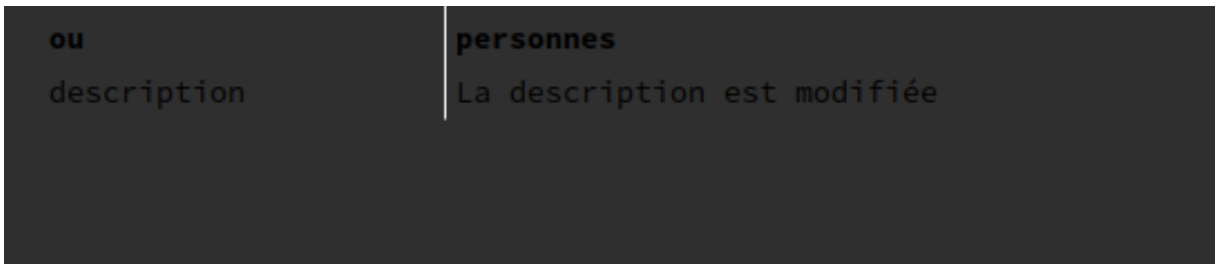
10) Nous allons changer l'attribut descrpition :

```
root@debian:/home/test/tp-ldap# cat description.ldif
dn: ou=personnes,dc=nodomain
changetype: modify
replace: description
description: La description est modifiée
```

Nous allons faire le ldapmodify :

```
root@debian:/home/test/tp-ldap# ldapmodify -v -x -D cn=admin,dc=
    nodomain -W -h localhost -f description.ldif
ldap_initialize( ldap://localhost )
Enter LDAP Password:
replace description:
    NOT ASCII (28 bytes)
modifying entry "ou=personnes,dc=nodomain"
modify complete
```

Nous pouvons voir sur apachedirectorystudio que la description est modifiée :



11) Il est possible d'utiliser l'UTF-8 à partir de la version 3 de LDAP.
12) A priori, la version utilisée est au moins la 3
13) Pour automatiser l'importation des fichiers ldif dans l'anuaire, nous allons pouvoir utiliser le script :

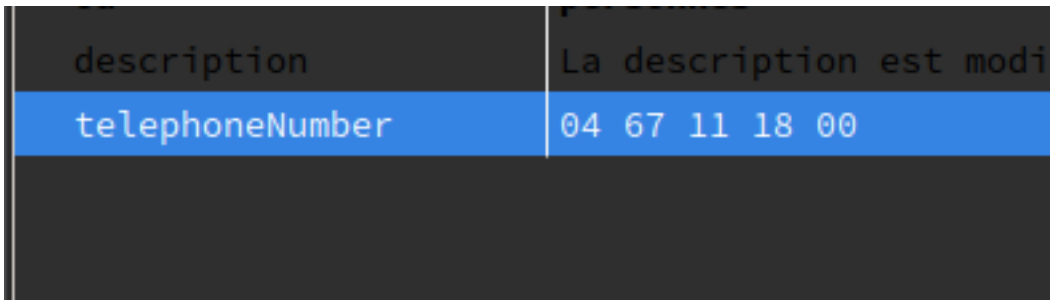
```
root@debian:/home/test/tp-ldap# cat ldif-import.sh
#!/bin/bash

# Le nom du fichier ldif sera à mettre en argument du script :

ldapmodify -v -x -D cn=admin,dc=nodomain -w root -h localhost -f $1
```

14) L'attribut à utiliser est : telephoneNumber
15) Nous allons utiliser le fichier .ldif suivant :

```
root@debian:/home/test/tp-ldap# cat ou-personnes-tel.ldif
dn: ou=personnes,dc=nodomain
changetype: modify
replace: telephoneNumber
telephoneNumber: 04 67 11 18 00
```

Nous pouvons importer le fichier à l'aide du script :

```
root@debian:/home/test/tp-ldap# ./ldif-import.sh ou-personnes-tel.ldif
ldap_initialize( ldap://localhost )
replace telephoneNumber:
    04 67 11 18 00
modifying entry "ou=personnes,dc=nodomain"
modify complete
```



16)

inetOrgPerson	organizationalPerson		• audio • businessCategory • carLicense • departmentNumber • displayName • employeeNumber • employeeType • givenName • homePhone • homePostalAddress • initials • jpegPhoto • labeledURI • mail • manager • mobile • o • pager • photo • roomNumber • secretary • uid • userCertificate • x500uniqueIdentifier • preferredLanguage • userSMIMECertificate • userPKCS12	The inetOrgPerson represents people who are associated with an organization in some way. It is a structural class and is derived from the organizationalPerson class which is defined in X.521.

17) Nous allons créer le fichier templates.txt suivant :

```
root@debian:/home/test/tp-ldap# cat templates.txt
dn: cn=*2,ou=personnes,dc=nodomain
changetype: add
objectclass: InetOrgPerson
cn: *2 *3
sn: *3
telephoneNumber: *4
mail: *5
description: *1
```

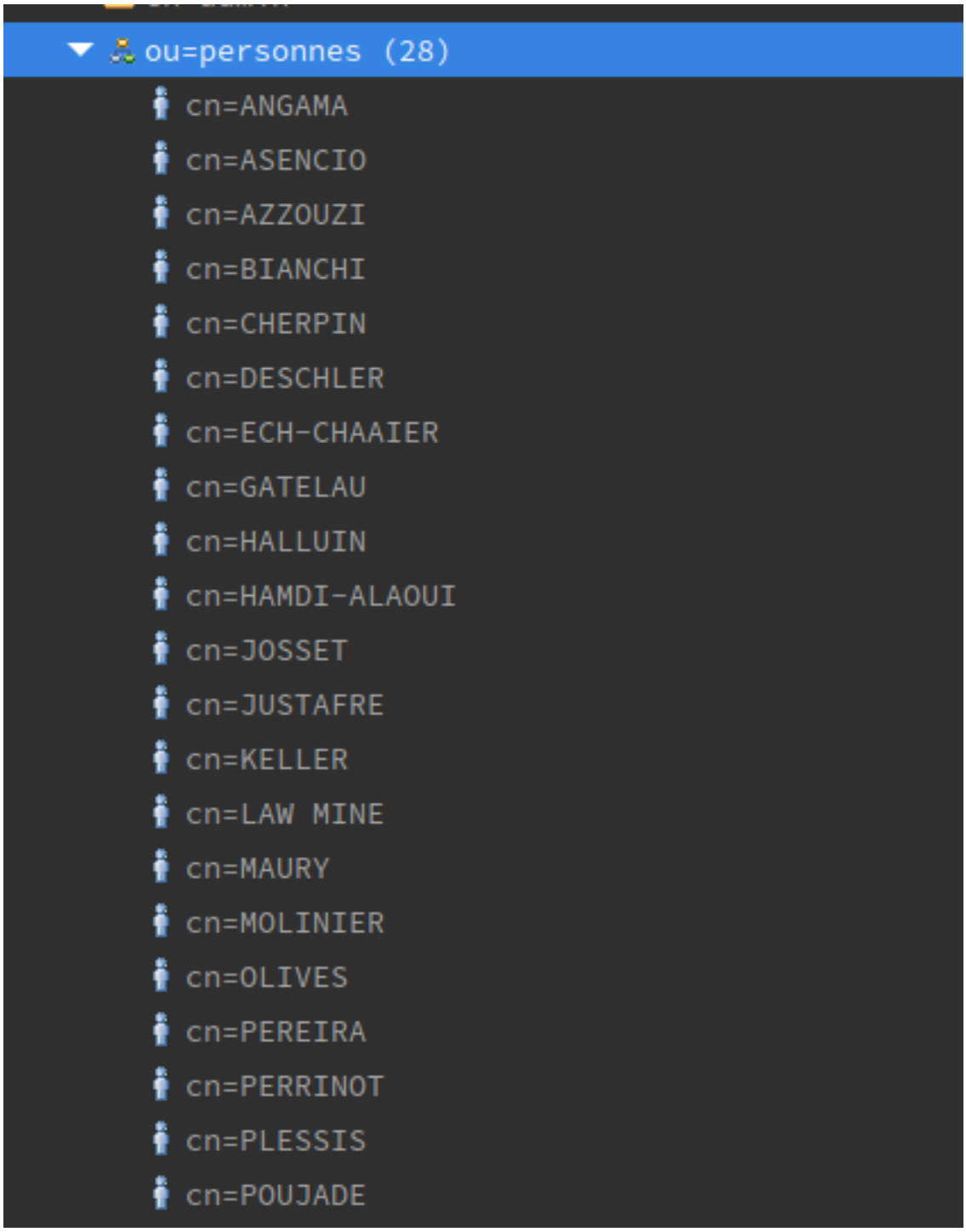
18) Nous allons exécuter le script pour créer le fichier makeldif.ldif :

```
root@debian:/home/test/tp-ldap# perl makeldif.pl -t templates.txt -a
    anciens-ISVD.csv >> makeldif.ldif
```

19) Le fichier makeldif.ldif est bien conforme aux normes, je vais donc l'importer dans l'annuaire :

```
root@debian:/home/test/tp-ldap# ./ldif-import.sh makeldif.ldif
```

Nous pouvons voir sur apachedirectorystudio que l'importation c'est bien passée :



20) A l'aide du man, nous allons expliquer la commande :

```
ldapsearch -LLL -x -b "dc=facerias,dc=org" "cn=admin"
-LLL : Permet de restreindre l'affichage au format LDIF, de supprimer
   les commentaires et de désactiver l'affichage de la version de LDIF
-x : utilise l'authentification simple plutôt que sasl
-b : défini le point de départ de la requête
"cn=admin" : objet que nous cherchons
```

21) Pour éviter de spécifier l'option -b, nous pouvons spécifier le Base DN dans ldap.conf

```
root@debian:/home/test/tp-ldap# cat /etc/ldap/ldap.conf
#
# LDAP Defaults
#

# See ldap.conf(5) for details
# This file should be world readable but not world writable.

#BASE   dc=example,dc=com
#URI    ldap://ldap.example.com ldap://ldap-master.example.com:666

#SIZELIMIT  12
#TIMELIMIT  15
#DEREF      never

# TLS certificates (needed for GnuTLS)
TLS_CACERT  /etc/ssl/certs/ca-certificates.crt
BASE    dc=nodomain
```

22) Cette commande :

```
root@debian:/home/test/tp-ldap# ldapsearch -LLL -x "(cn=admin)"
dn: cn=admin,dc=nodomain
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator
```

recherche l'objet admin.

23)

```
root@debian:/home/test/tp-ldap# ldapsearch -LLL -x "(cn=admin)"
Recherche toute la classe admin

root@debian:/home/test/tp-ldap# ldapsearch -LLL -x "(cn=admin)" dn
Recherche uniquement le dn de la classe admin

root@debian:/home/test/tp-ldap# ldapsearch -LLL -x "(cn=admin)" cn
Recherche seulement le cn de la classe admin

root@debian:/home/test/tp-ldap# ldapsearch -LLL -x "(cn=admin)" dn cn
Recherche le cn et le dn de la classe admin
```

24)

```
root@debian:/home/test/tp-ldap# ldapsearch -LLL -s base -x dn
dn: dc=nodomain
Cette commande sort le dn depuis la base définie dans ldap.conf

root@debian:/home/test/tp-ldap# ldapsearch -LLL -s onelevel -x dn
dn: cn=admin,dc=nodomain

dn: ou=personnes,dc=nodomain
Cette commande va partir de la base, descendre d'un étage dans l'
   annuaire et retourner les dn correspondant pour chaque classe
```

25)

```
ldapsearch -LLL -s children -x "(objectClass=inetOrgPerson)"
```

```
ldapsearch -LLL -s children -x "(description=2008)"
```

```
ldapsearch -LLL -s children -x "(sn=B*)"
```

```
ldapsearch -LLL -s children -x "(|(sn=B*)(sn=C*))"
```

26)

```
ldapsearch -LLL -s children -x "(sn=A*)"
```

```
echo 'QU5HQU1BIFN0w6lwaGFuZQ==' | base64 -d
          ANGAMA Stéphane
```

27)

```
ldapsearch -LLL -s children -x "(sn=C*)" +
```

28)

```
ldapsearch -LLL -s children -x "(sn=C*)" CreateTimeStamp  CreatorsName
```