TABLE DES MATIÈRES

Jean-François (Jeff) **ANDRIATSIMEVA** - Moussa **CAMARA** Rachel **KOEHLER** - Thomas **VERAN**

Job 1 : Comprendre et préparer LDAP

1. Présentation

Effectuez une recherche sur LDAP et rédigez une courte présentation de ce qu'est LDAP, son utilité dans les entreprises, et les avantages de centraliser la gestion des accès.

2. Définir les Groupes d'Utilisateurs

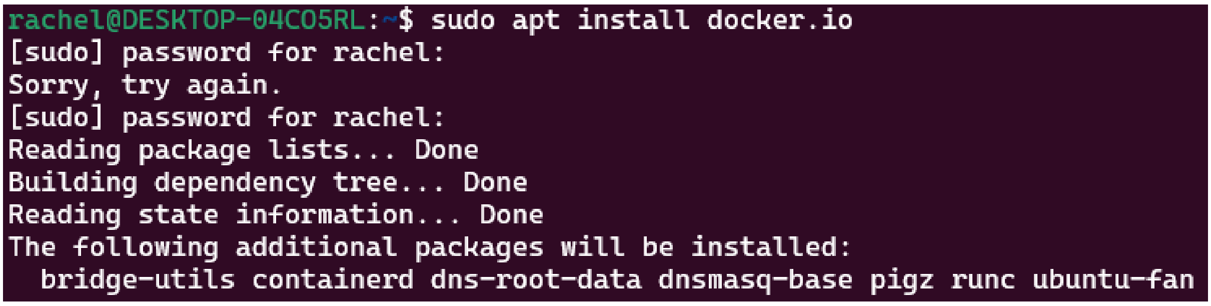
Proposez une organisation simple des utilisateurs dans l'entreprise selon trois types de rôles :

- Utilisateurs basiques : Ont accès aux fichiers partagés.
- Développeurs : Ont un accès limité aux applications de développement.
- Administrateurs : Ont tous les droits, y compris la modification des paramètres de sécurité.
***Livrable** : Une liste des groupes avec une brève explication des droits pour chaque groupe.*

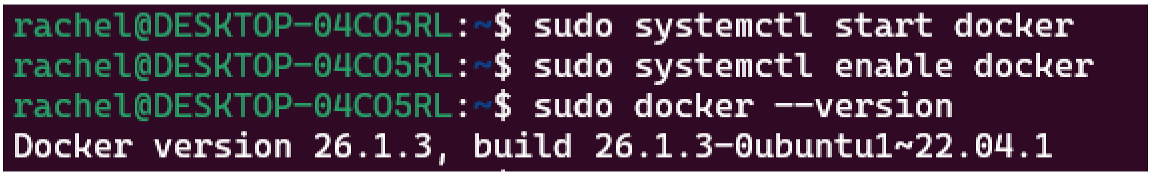
Job 2 : Mise en place technique

1. Installer un Serveur LDAP

Suivez un tutoriel pour installer et configurer un serveur LDAP basique sur un système Linux. Notez chaque étape pour comprendre le processus.
Installation & démarrage de Docker sur Ubuntu

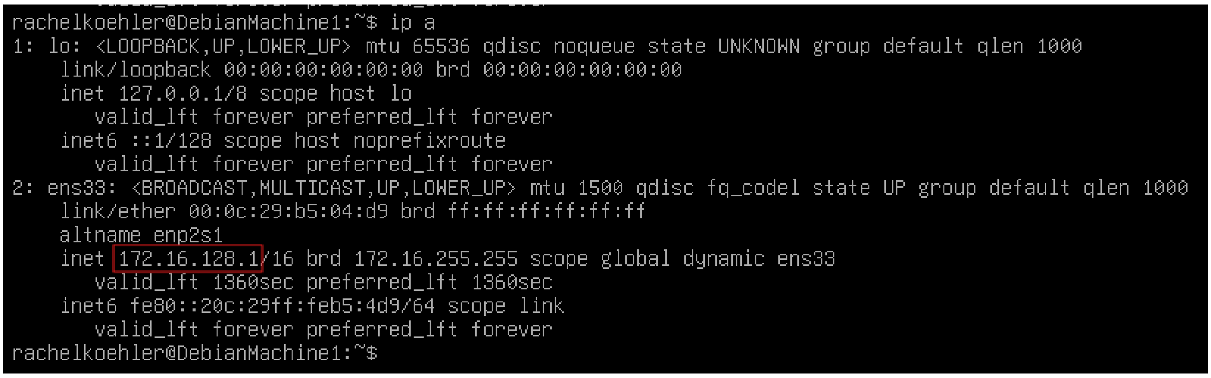


```
rachel@DESKTOP-04CO5RL:~$ sudo apt install docker.io
[sudo] password for rachel:
Sorry, try again.
[sudo] password for rachel:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan
```

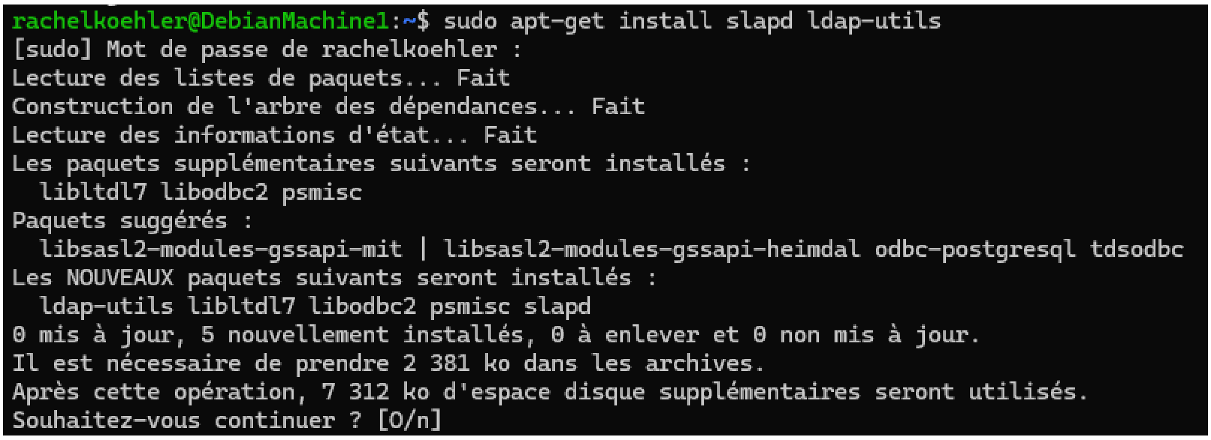


```
rachel@DESKTOP-04CO5RL:~$ sudo systemctl start docker
rachel@DESKTOP-04CO5RL:~$ sudo systemctl enable docker
rachel@DESKTOP-04CO5RL:~$ sudo docker --version
Docker version 26.1.3, build 26.1.3-0ubuntu1~22.04.1
```

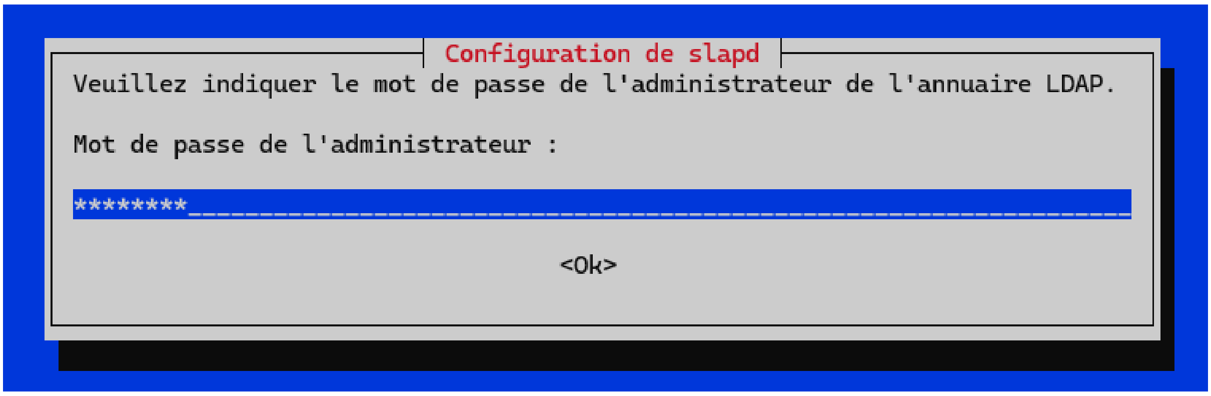
SSH -> ip vm



```
rachelkoehler@DebianMachine1:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:b5:04:d9 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.16.128.1/16 brd 172.16.255.255 scope global dynamic ens33
       valid_lft 1360sec preferred_lft 1360sec
    inet6 fe80::20c:29ff:feb5:4d9/64 scope link
       valid_lft forever preferred_lft forever
rachelkoehler@DebianMachine1:~$
```

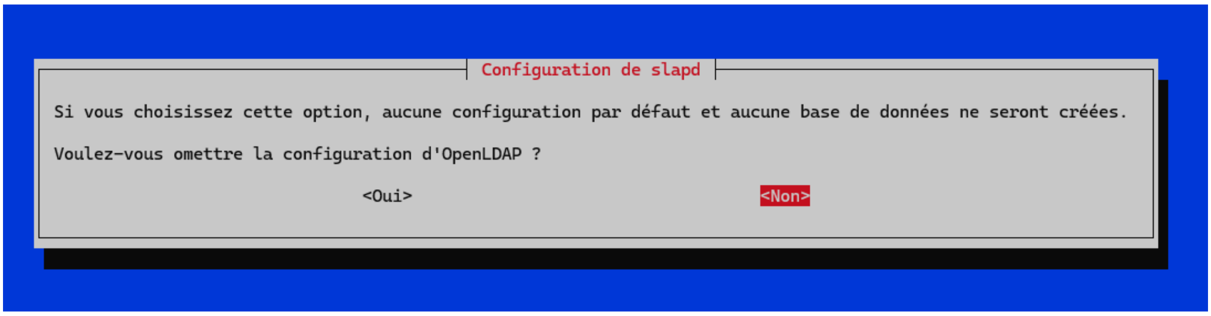


```
rachelkoehler@DebianMachine1:~$ sudo apt-get install slapd ldap-utils
[sudo] Mot de passe de rachelkoehler :
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  libltdl7 libodbc2 psmisc
Paquets suggérés :
  libsasl2-modules-gssapi-mit | libsasl2-modules-gssapi-heimdal odbc-postgresql tdsodbc
Les NOUVEAUX paquets suivants seront installés :
  ldap-utils libltdl7 libodbc2 psmisc slapd
0 mis à jour, 5 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 2 381 ko dans les archives.
Après cette opération, 7 312 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n]
```

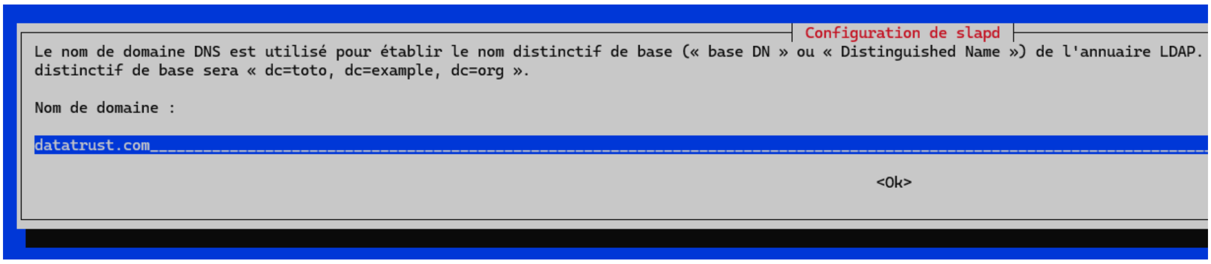


```
Configuration de slapd
Veuillez indiquer le mot de passe de l'administrateur de l'annuaire LDAP.

Mot de passe de l'administrateur :

********_______________________________________________________________

<Ok>
```

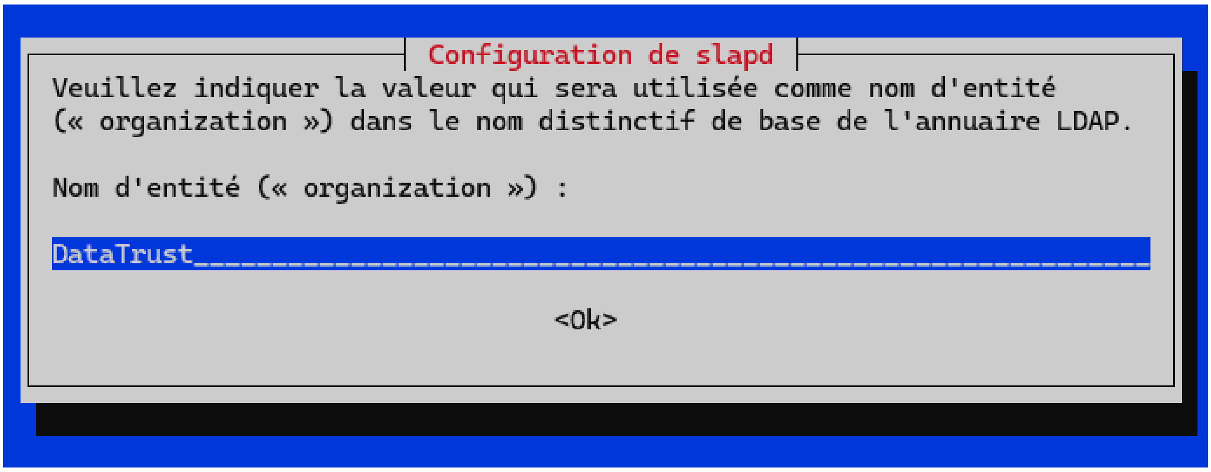
sudo dpkg-reconfigure slapd



```
Si vous choisissez cette option, aucune configuration par défaut et aucune base de données ne seront créées.

Voulez-vous omettre la configuration d'OpenLDAP ?
```

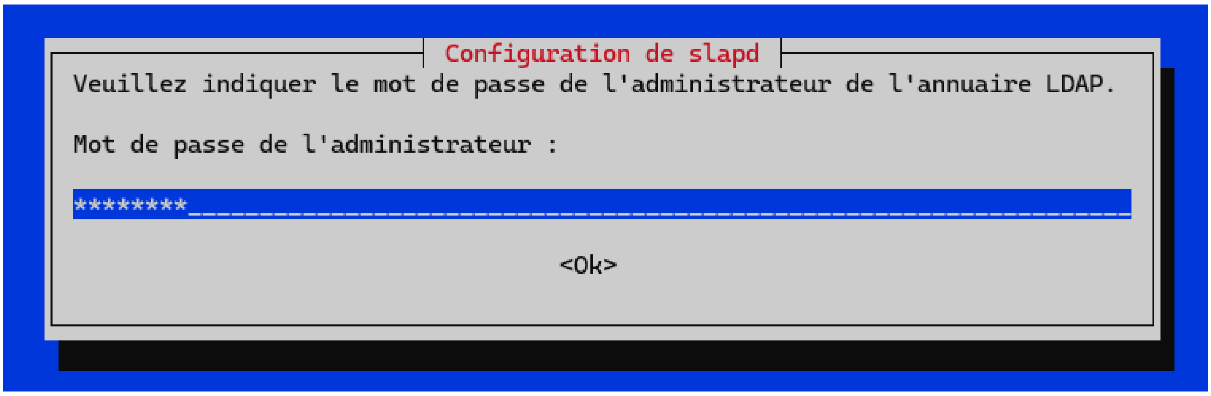


```
Le nom de domaine DNS est utilisé pour établir le nom distinctif de base (« base DN » ou « Distinguished Name ») de l'annuaire LDAP.
distinctif de base sera « dc=toto, dc=example, dc=org ».

Nom de domaine :

datatrust.com_
```

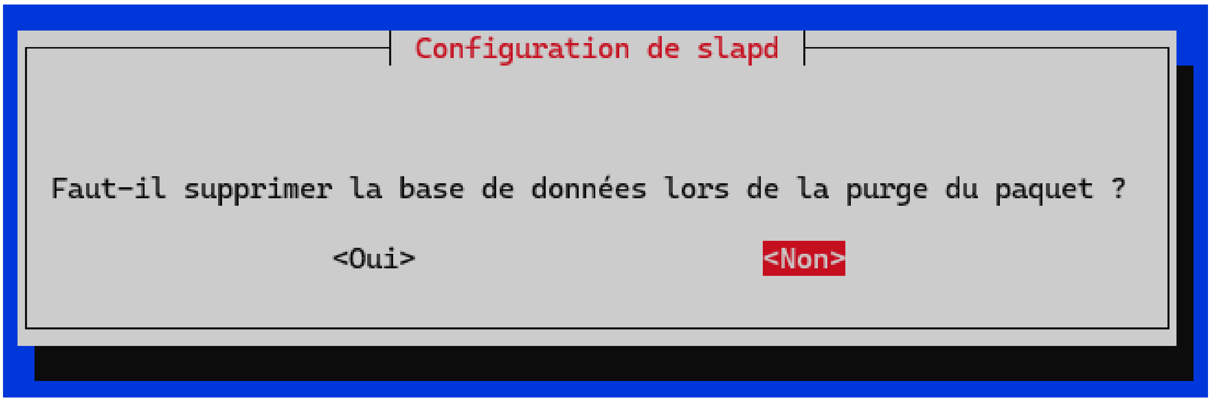


```
Veuillez indiquer la valeur qui sera utilisée comme nom d'entité
(« organization ») dans le nom distinctif de base de l'annuaire LDAP.

Nom d'entité (« organization ») :

DataTrust_
```

mdp = adminmdp

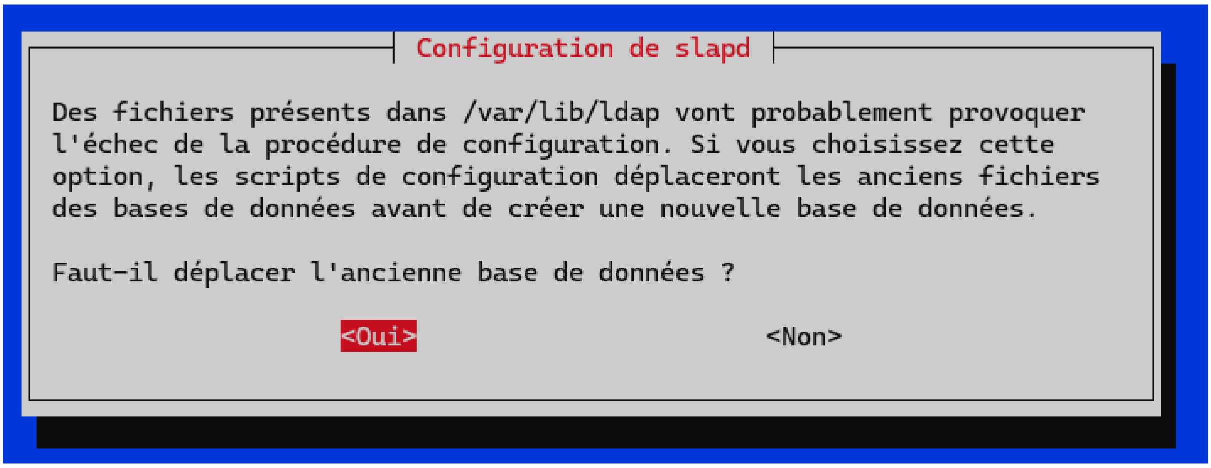


```
Veuillez indiquer le mot de passe de l'administrateur de l'annuaire LDAP.

Mot de passe de l'administrateur :

********_
```



Configuration de slapd

Faut-il supprimer la base de données lors de la purge du paquet ?

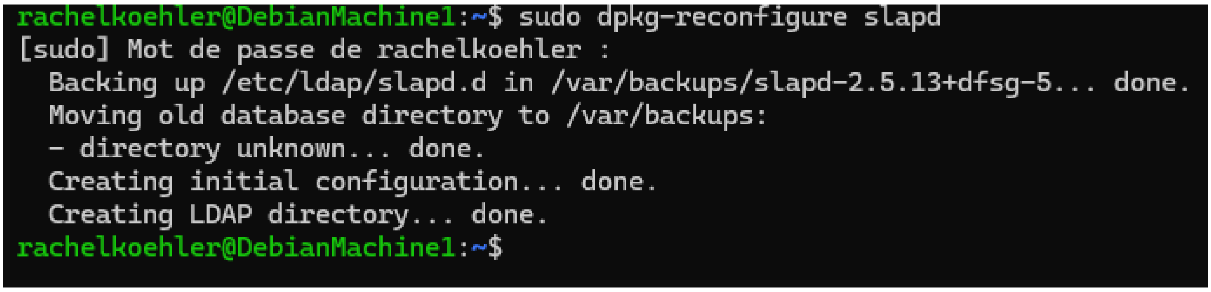


Configuration de slapd

Des fichiers présents dans /var/lib/ldap vont probablement provoquer l'échec de la procédure de configuration. Si vous choisissez cette option, les scripts de configuration déplaceront les anciens fichiers des bases de données avant de créer une nouvelle base de données.

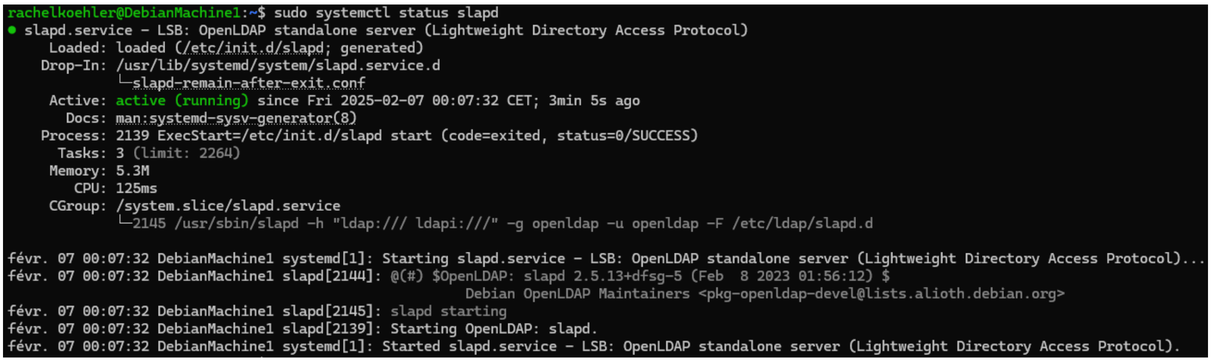
Faut-il déplacer l'ancienne base de données ?

Fin configuration fichier slapd



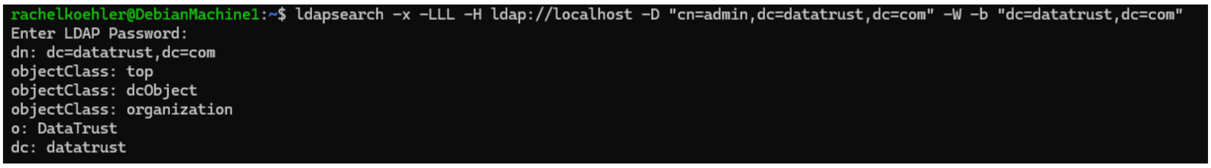
```
rachelkoehler@DebianMachine1:~$ sudo dpkg-reconfigure slapd
[sudo] Mot de passe de rachelkoehler :
  Backing up /etc/ldap/slapd.d in /var/backups/slapd-2.5.13+dfsg-5... done.
  Moving old database directory to /var/backups:
  - directory unknown... done.
  Creating initial configuration... done.
  Creating LDAP directory... done.
rachelkoehler@DebianMachine1:~$
```

Vérifier la bonne configuration/installation du serveur LDAP



```
rachelkoehler@DebianMachine1:~$ sudo systemctl status slapd
● slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol)
     Loaded: loaded (/etc/init.d/slapd; generated)
    Drop-In: /usr/lib/systemd/system/slapd.service.d
             └─slapd-remain-after-exit.conf
     Active: active (running) since Fri 2025-02-07 00:07:32 CET; 3min 5s ago
       Docs: man:systemd-sysv-generator(8)
    Process: 2139 ExecStart=/etc/init.d/slapd start (code=exited, status=0/SUCCESS)
      Tasks: 3 (limit: 2264)
     Memory: 5.3M
        CPU: 125ms
     CGroup: /system.slice/slapd.service
             └─2145 /usr/sbin/slapd -h "ldap:/// ldapi:///" -g openldap -u openldap -F /etc/ldap/slapd.d

févr. 07 00:07:32 DebianMachine1 systemd[1]: Starting slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol)...
févr. 07 00:07:32 DebianMachine1 slapd[2144]: @(#) $OpenLDAP: slapd 2.5.13+dfsg-5 (Feb  8 2023 01:56:12) $
                                                       Debian OpenLDAP Maintainers <pkg-openldap-devel@lists.alioth.debian.org>
févr. 07 00:07:32 DebianMachine1 slapd[2145]: slapd starting
févr. 07 00:07:32 DebianMachine1 slapd[2139]: Starting OpenLDAP: slapd.
févr. 07 00:07:32 DebianMachine1 systemd[1]: Started slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol).
```

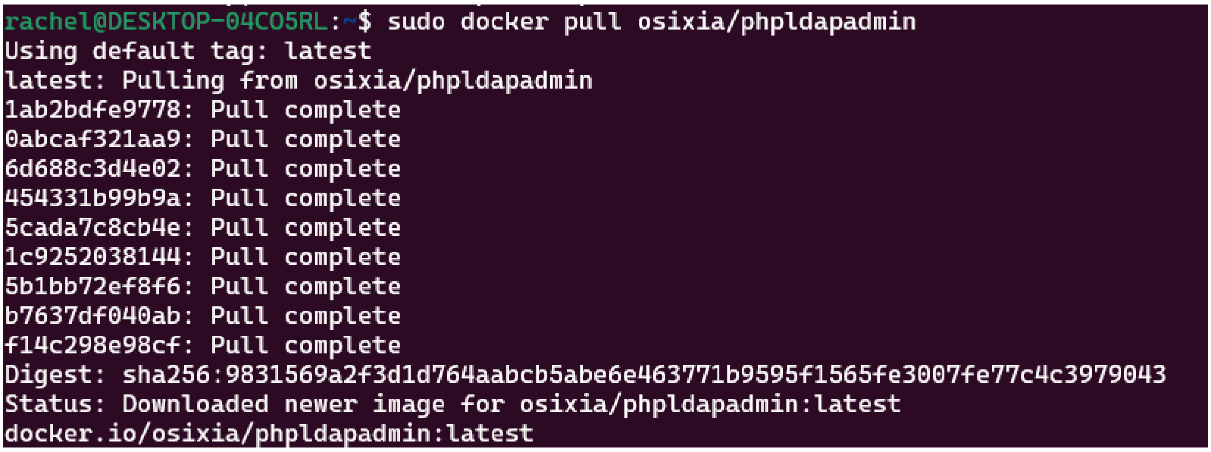
Vérifier si le LDAP est configuré proprement



```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -LLL -H ldap://localhost -D "cn=admin,dc=datatrust,dc=com" -W -b "dc=datatrust,dc=com"
Enter LDAP Password:
dn: dc=datatrust,dc=com
objectClass: top
objectClass: dcObject
objectClass: organization
o: DataTrust
dc: datatrust
```

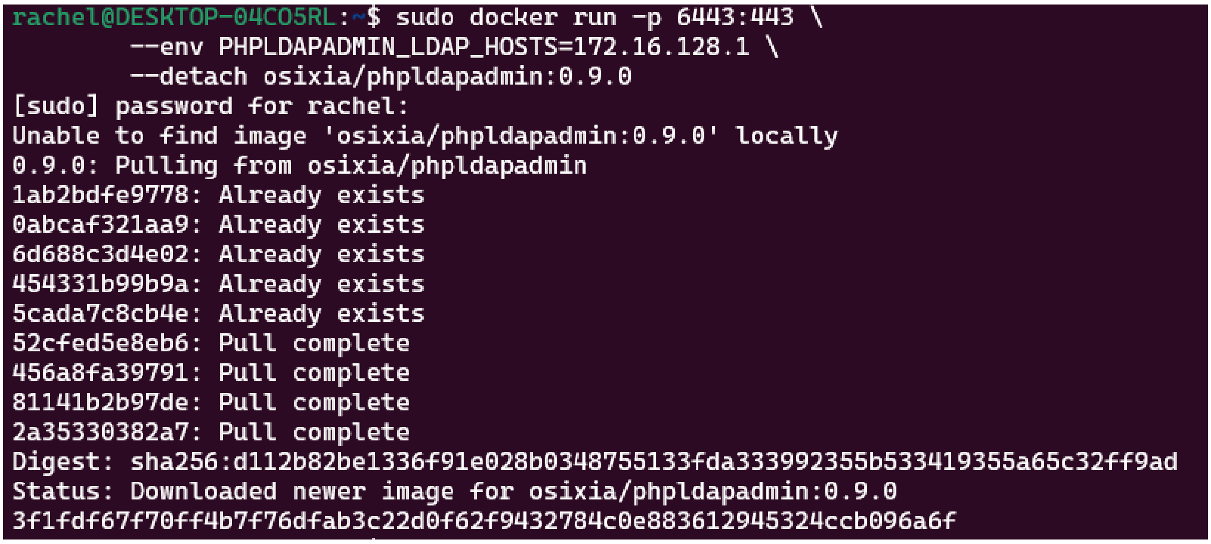
Installer phpLDAPadmin

1.Tirez l'image Docker phpLDAPadmin :



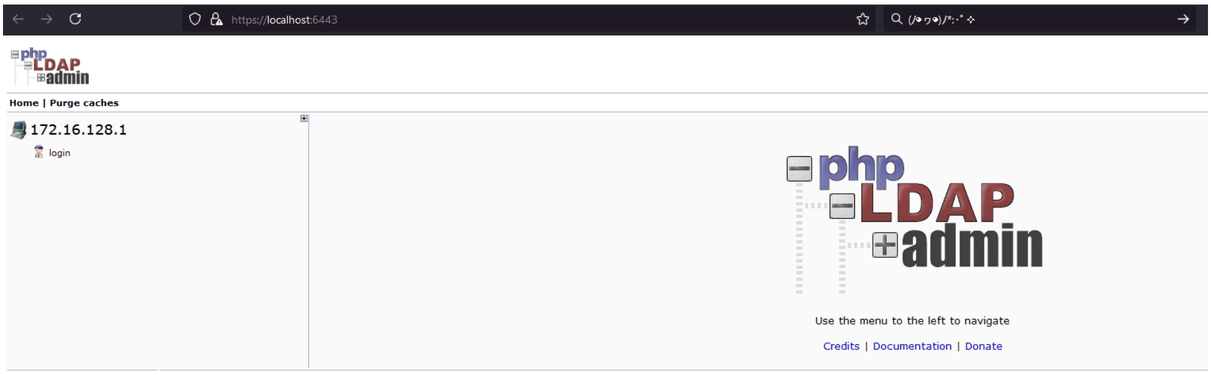
```
rachel@DESKTOP-04CO5RL:~$ sudo docker pull osixia/phpldapadmin
Using default tag: latest
latest: Pulling from osixia/phpldapadmin
1ab2bdfe9778: Pull complete
0abcaf321aa9: Pull complete
6d688c3d4e02: Pull complete
454331b99b9a: Pull complete
5cada7c8cb4e: Pull complete
1c9252038144: Pull complete
5b1bb72ef8f6: Pull complete
b7637df040ab: Pull complete
f14c298e98cf: Pull complete
Digest: sha256:9831569a2f3d1d764aabcb5abe6e463771b9595f1565fe3007fe77c4c3979043
Status: Downloaded newer image for osixia/phpldapadmin:latest
docker.io/osixia/phpldapadmin:latest
```

2. Lancez un conteneur phpLDAPadmin :



```
rachel@DESKTOP-04CO5RL:~$ sudo docker run -p 6443:443 \
        --env PHPLDAPADMIN_LDAP_HOSTS=172.16.128.1 \
        --detach osixia/phpldapadmin:0.9.0
[sudo] password for rachel:
Unable to find image 'osixia/phpldapadmin:0.9.0' locally
0.9.0: Pulling from osixia/phpldapadmin
1ab2bdfe9778: Already exists
0abcaf321aa9: Already exists
6d688c3d4e02: Already exists
454331b99b9a: Already exists
5cada7c8cb4e: Already exists
52cfed5e8eb6: Pull complete
456a8fa39791: Pull complete
81141b2b97de: Pull complete
2a35330382a7: Pull complete
Digest: sha256:d112b82be1336f91e028b0348755133fda333992355b533419355a65c32ff9ad
Status: Downloaded newer image for osixia/phpldapadmin:0.9.0
3f1fdf67f70ff4b7f76dfab3c22d0f62f9432784c0e883612945324ccb096a6f
```

3. Accéder à phpLDAPadmin

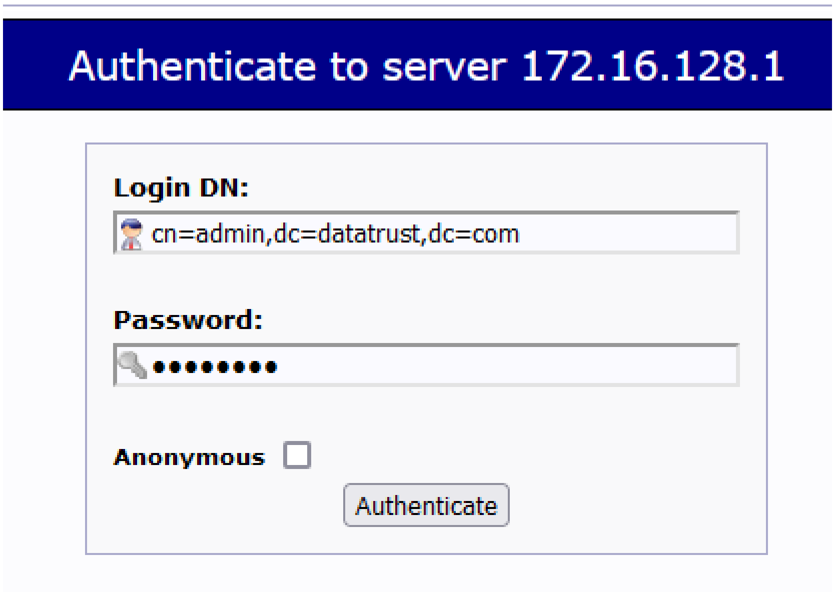


Use the menu to the left to navigate

- Ouvrir le navigateur et aller à : `https://localhost:6443`
- Utiliser les identifiants :

  - Connectez-vous avec les identifiants suivants :
    - Login DN : `cn=admin,dc=datatrust,dc=com`
    - Mot de passe : `adminpassword` (celui défini précédemment).

Ce qui donne:



Authenticate to server 172.16.128.1

Login DN:

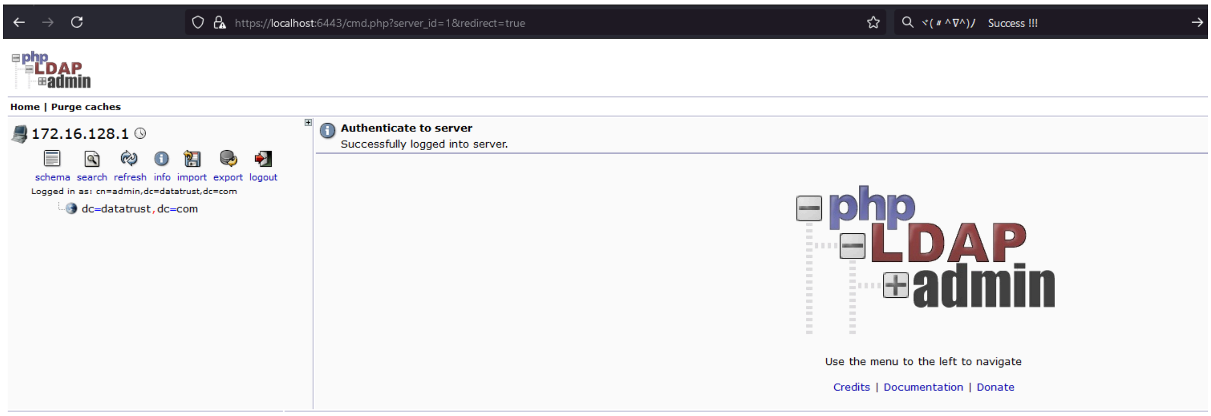
cn=admin,dc=datatrust,dc=com

Password:

••••••••

Anonymous □

Authenticate



https://localhost:6443/cmd.php?server_id=1&redirect=true

Success !!!

Home | Purge caches

172.16.128.1

Logged in as: cn=admin,dc=datatrust,dc=com

dc=datatrust, dc=com

Authenticate to server

Successfully logged into server.

Use the menu to the left to navigate

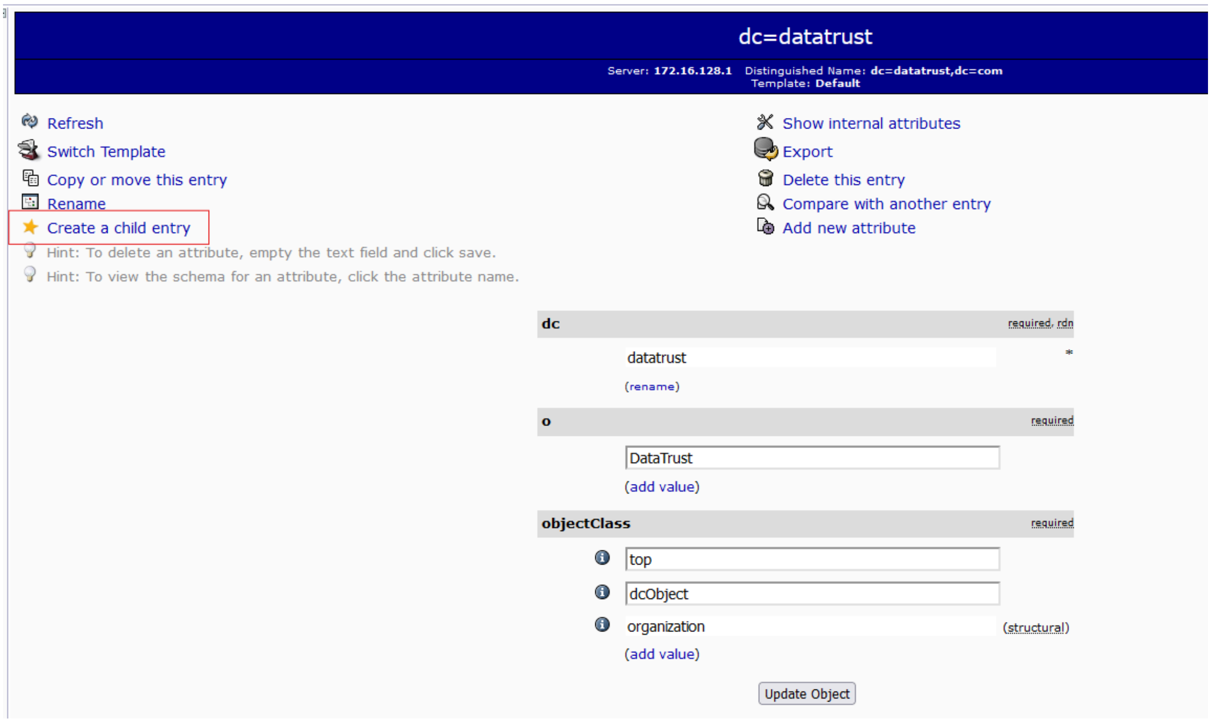
Credits | Documentation | Donate

2. Créer des Utilisateurs

Créez trois utilisateurs (un pour chaque groupe défini) et attribuez-les à leurs groupes respectifs. Par exemple :
- Alice : Utilisateur basique
- Bob : Développeur
- Claire : Administratrice

Comment créer un groupe (ici 3) et un user (ici 3 aussi) ?

CRÉATION DES GROUPES



dc=datatrust

Server: **172.16.128.1** Distinguished Name: **dc=datatrust,dc=com**
Template: **Default**

Hint: To delete an attribute, empty the text field and click save.

Hint: To view the schema for an attribute, click the attribute name.

dc required, rdn

datatrust *

(rename)

o required

DataTrust

(add value)

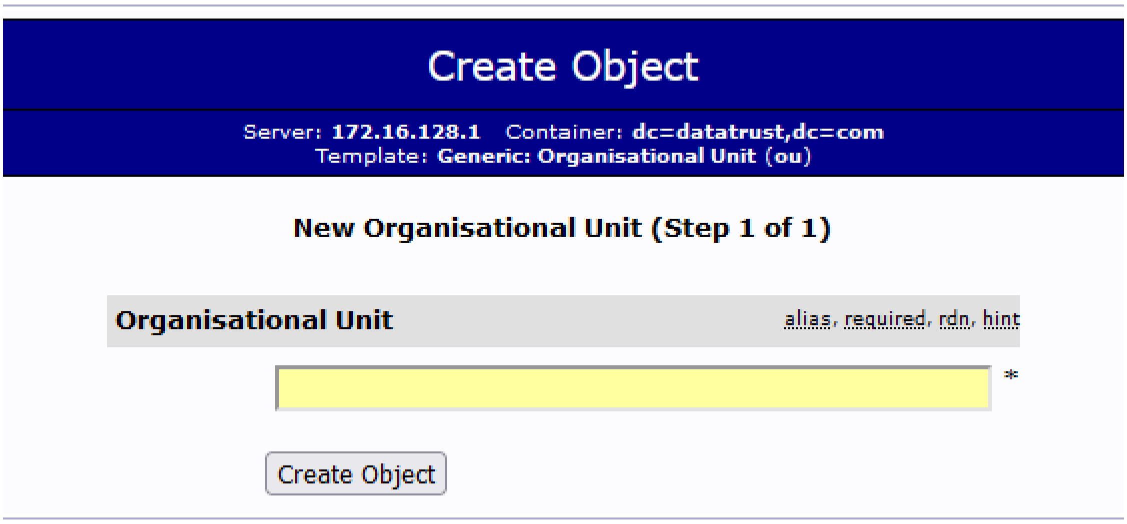
objectClass required

top

dcObject

organization (structural)

(add value)



Create Object

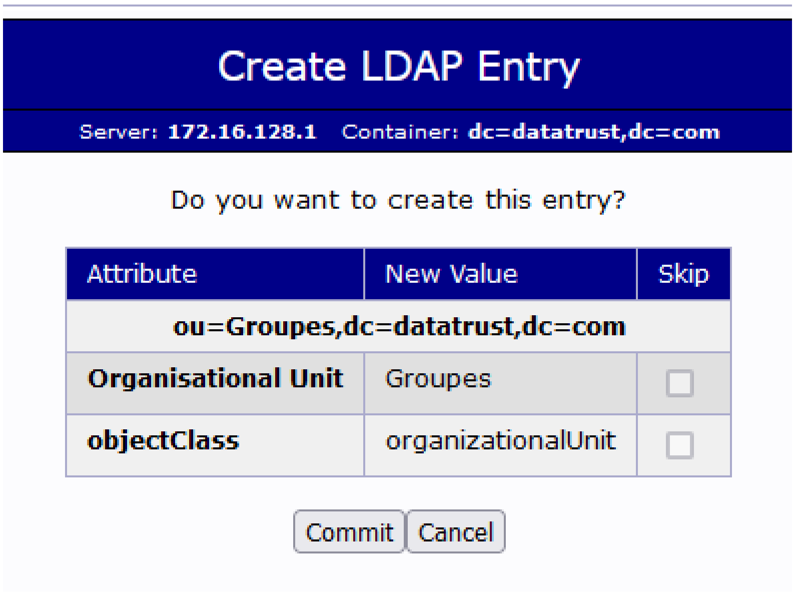
Server: **172.16.128.1** Container: **dc=datatrust,dc=com**
Template: **Generic: Organisational Unit (ou)**

New Organisational Unit (Step 1 of 1)

Organisational Unit alias, required, rdn, hint

*

Création de l’ou “groupes”:



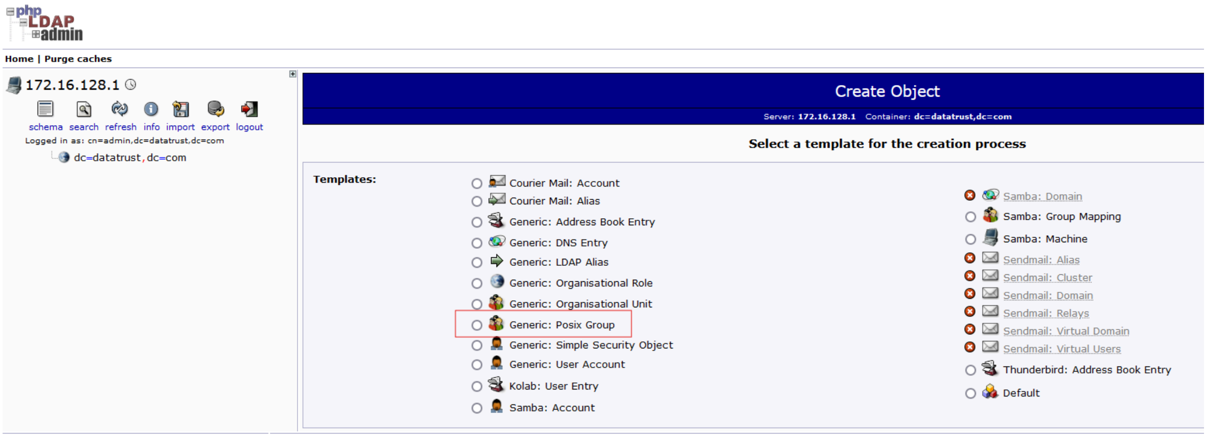
Create LDAP Entry

Server: **172.16.128.1** Container: **dc=datatrust,dc=com**

Do you want to create this entry?

Attribute	New Value	Skip
ou=Groupes,dc=datatrust,dc=com		
Organisational Unit	Groupes	□
objectClass	organizationalUnit	□

Choisir le modèle Posixgroup pour définir les gorupes (basicuser, developer, admin)



Logged in as: cn=admin,dc=datatrust,dc=com

dc=datatrust,dc=com

Create Object

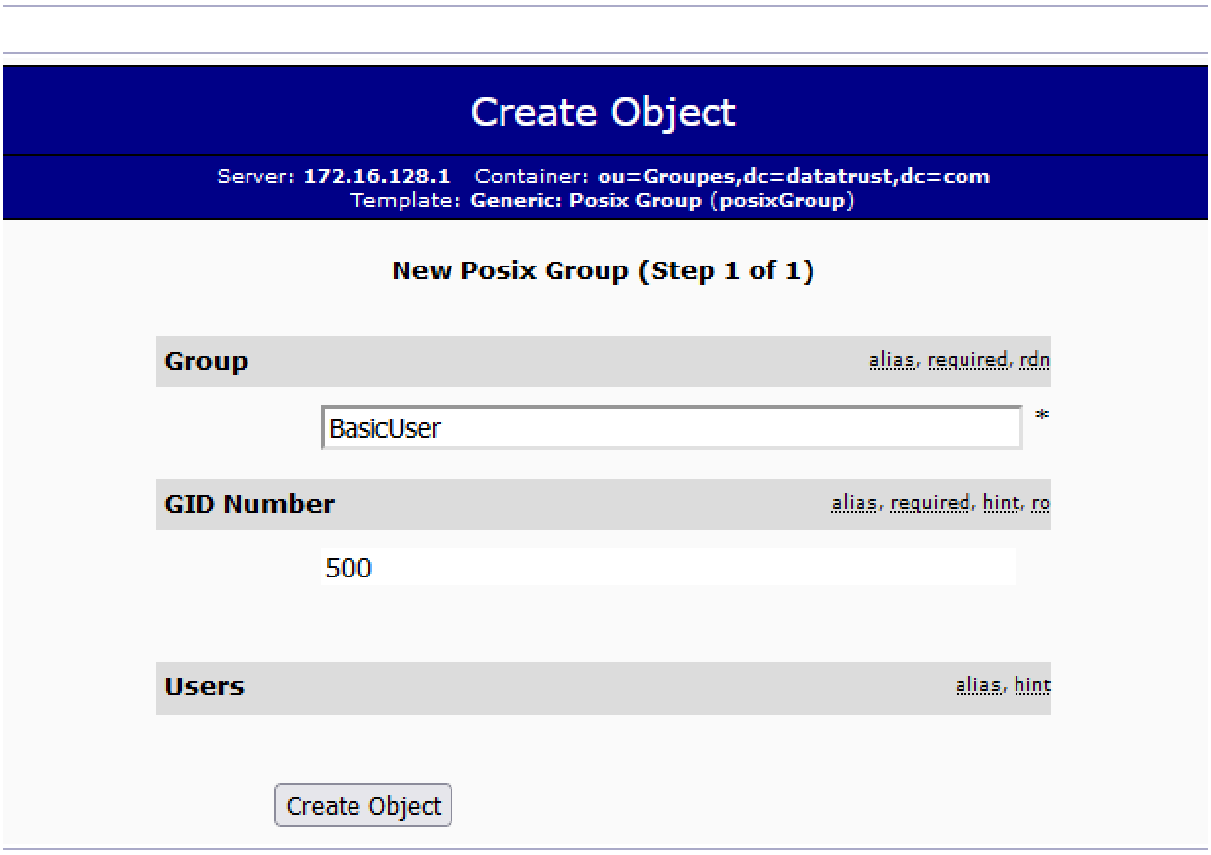
Server: **172.16.128.1** Container: **dc=datatrust,dc=com**

Select a template for the creation process

Templates:

○ Courier Mail: Account
○ Courier Mail: Alias
○ Generic: Address Book Entry
○ Generic: DNS Entry
○ Generic: LDAP Alias
○ Generic: Organisational Role
○ Generic: Organisational Unit
○ Generic: Posix Group
○ Generic: Simple Security Object
○ Generic: User Account
○ Kolab: User Entry
○ Samba: Account
☒ Samba: Domain
○ Samba: Group Mapping
○ Samba: Machine
☒ Sendmail: Alias
☒ Sendmail: Cluster
☒ Sendmail: Domain
☒ Sendmail: Relays
☒ Sendmail: Virtual Domain
☒ Sendmail: Virtual Users
○ Thunderbird: Address Book Entry
○ Default

Pour “Basic Users”



Create Object

Server: **172.16.128.1** Container: **ou=Groupes,dc=datatrust,dc=com**
Template: **Generic: Posix Group (posixGroup)**

New Posix Group (Step 1 of 1)

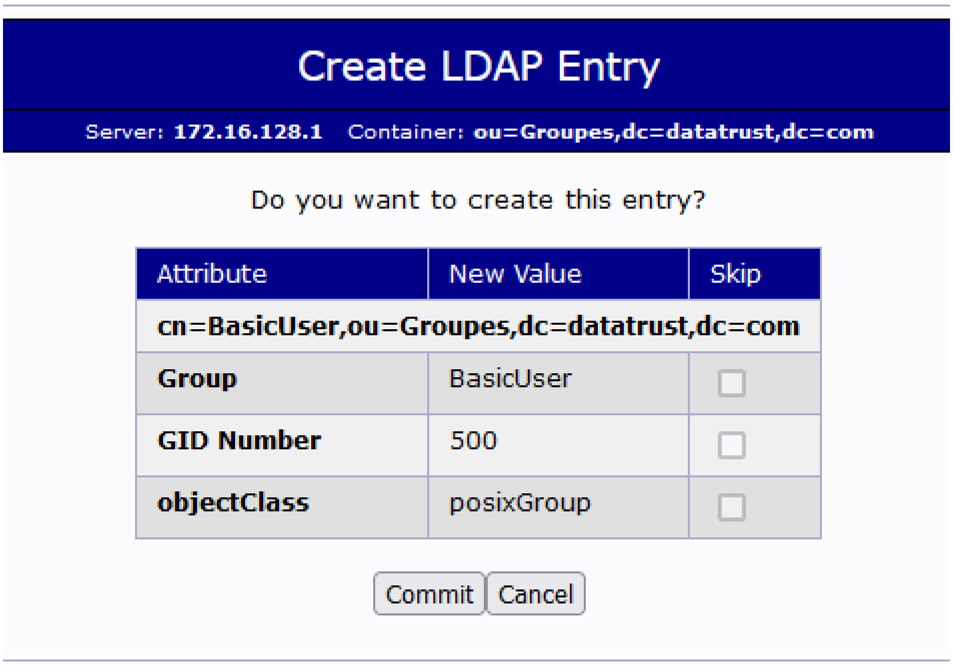
Group alias, required, rdn

BasicUser *

GID Number alias, required, hint, ro

500

Users alias, hint



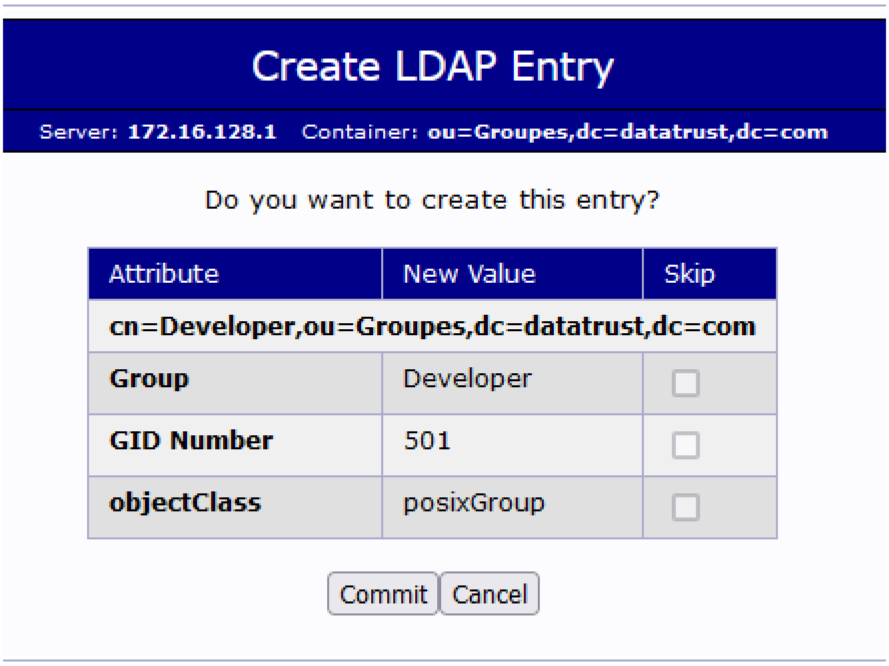
Create LDAP Entry

Server: **172.16.128.1** Container: **ou=Groupes,dc=datatrust,dc=com**

Do you want to create this entry?

Attribute	New Value	Skip
cn=BasicUser,ou=Groupes,dc=datatrust,dc=com		
Group	BasicUser	□
GID Number	500	□
objectClass	posixGroup	□

Pour “Dvlopr”



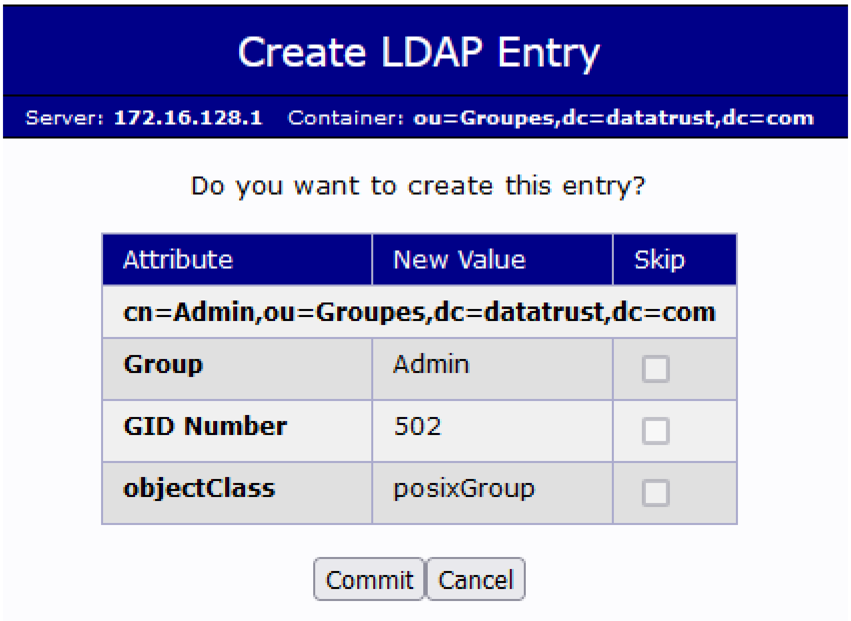
Create LDAP Entry

Server: **172.16.128.1** Container: **ou=Groupes,dc=datatrust,dc=com**

Do you want to create this entry?

Attribute	New Value	Skip
cn=Developer,ou=Groupes,dc=datatrust,dc=com		
Group	Developer	□
GID Number	501	□
objectClass	posixGroup	□

Pour “Admin”



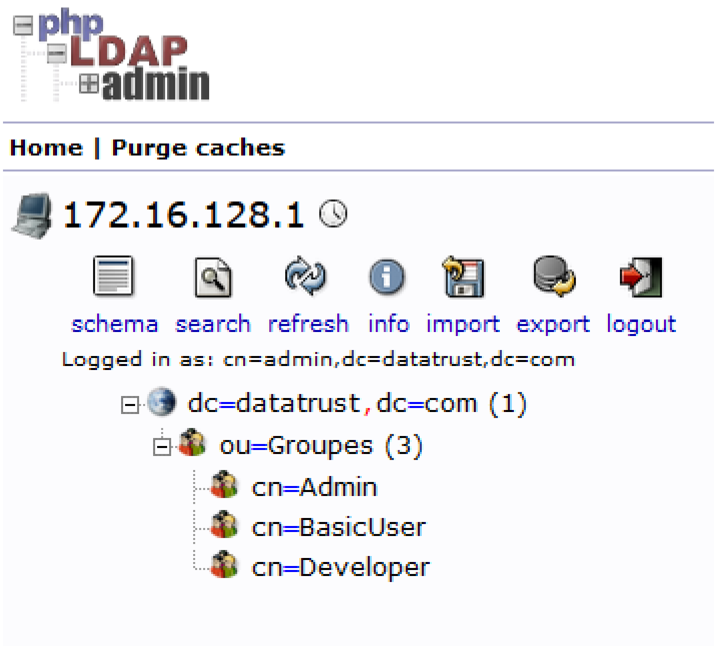
Create LDAP Entry

Server: **172.16.128.1** Container: **ou=Groupes,dc=datatrust,dc=com**

Do you want to create this entry?

Attribute	New Value	Skip
cn=Admin,ou=Groupes,dc=datatrust,dc=com		
Group	Admin	□
GID Number	502	□
objectClass	posixGroup	□

CCL groupes:

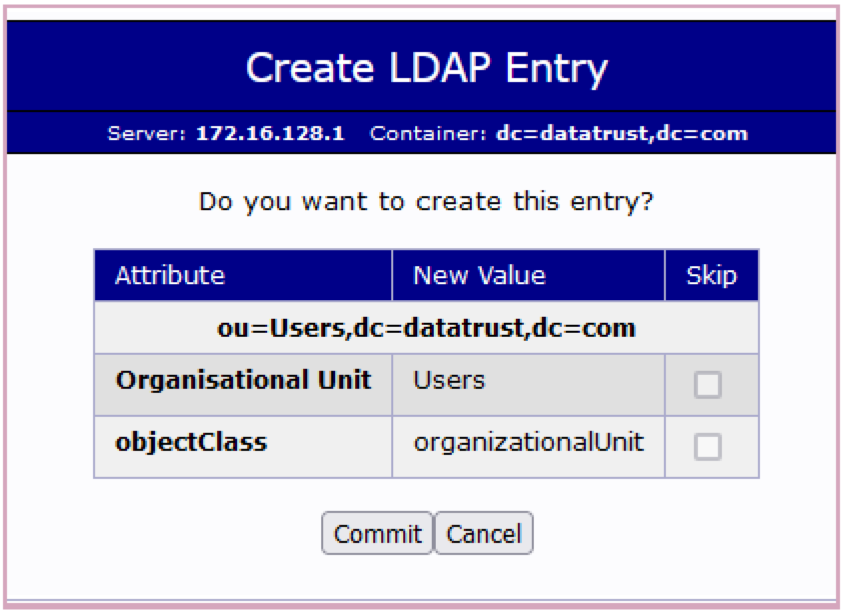


172.16.128.1

Logged in as: cn=admin,dc=datatrust,dc=com

```
⊟ dc=datatrust,dc=com (1)
  ⊟ ou=Groupes (3)
      cn=Admin
      cn=BasicUser
      cn=Developer
```

CRÉATION DES USERS

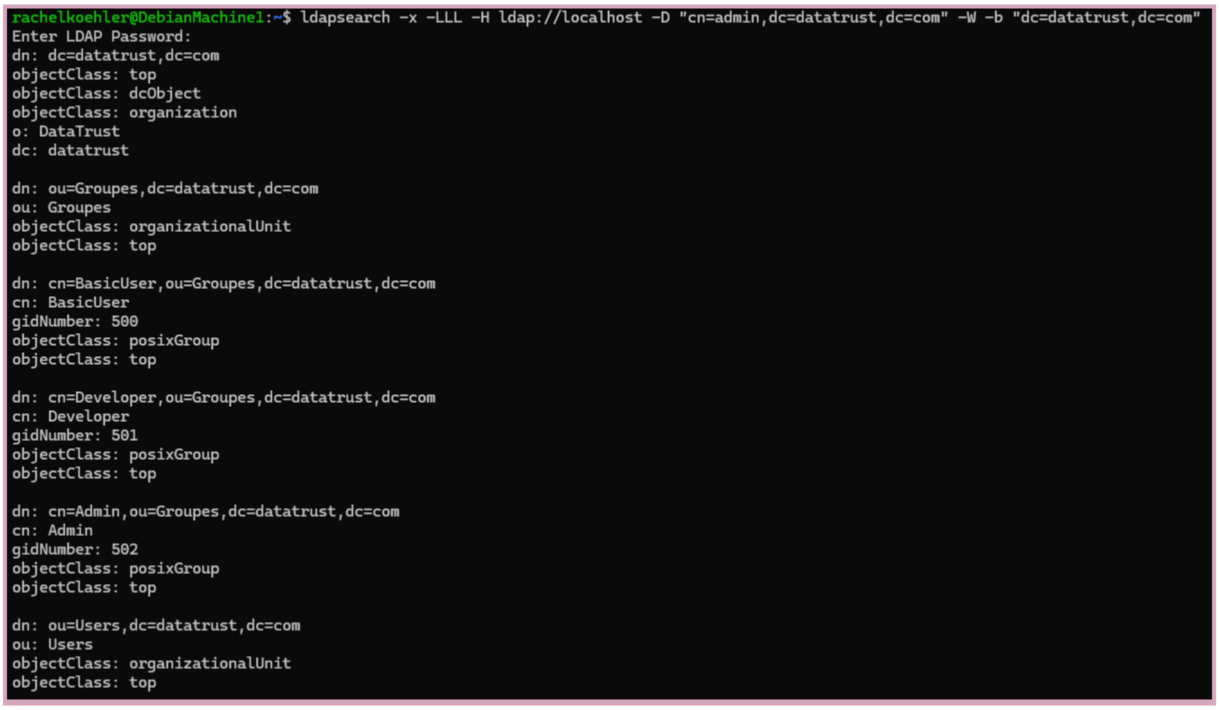


Create LDAP Entry

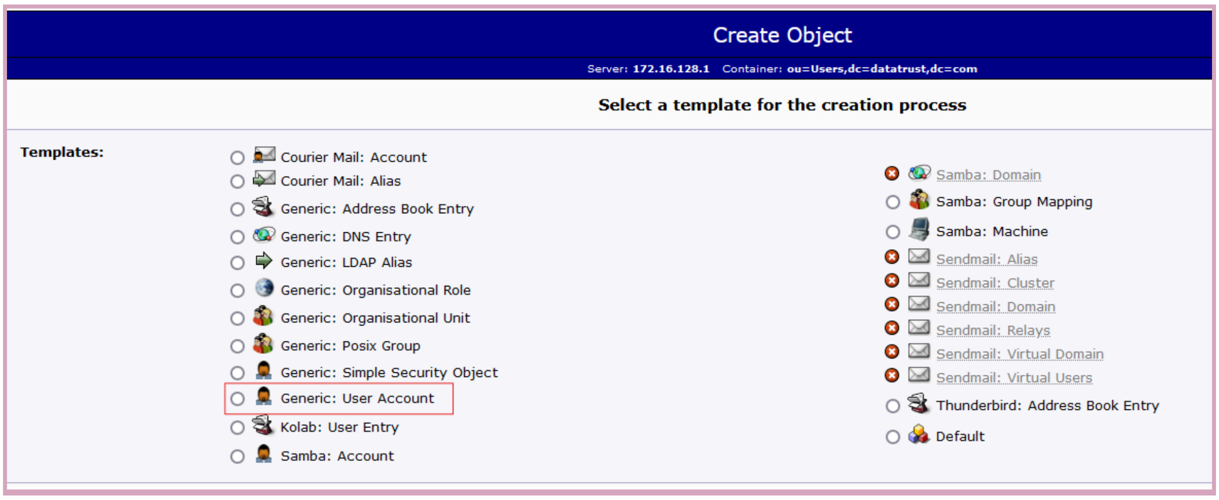
Server: **172.16.128.1** Container: **dc=datatrust,dc=com**

Do you want to create this entry?

Attribute	New Value	Skip
ou=Users,dc=datatrust,dc=com		
Organisational Unit	Users	□
objectClass	organizationalUnit	□



```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -LLL -H ldap://localhost -D "cn=admin,dc=datatrust,dc=com" -W -b "dc=datatrust,dc=com"
Enter LDAP Password:
dn: dc=datatrust,dc=com
objectClass: top
objectClass: dcObject
objectClass: organization
o: DataTrust
dc: datatrust

dn: ou=Groupes,dc=datatrust,dc=com
ou: Groupes
objectClass: organizationalUnit
objectClass: top

dn: cn=BasicUser,ou=Groupes,dc=datatrust,dc=com
cn: BasicUser
gidNumber: 500
objectClass: posixGroup
objectClass: top

dn: cn=Developer,ou=Groupes,dc=datatrust,dc=com
cn: Developer
gidNumber: 501
objectClass: posixGroup
objectClass: top

dn: cn=Admin,ou=Groupes,dc=datatrust,dc=com
cn: Admin
gidNumber: 502
objectClass: posixGroup
objectClass: top

dn: ou=Users,dc=datatrust,dc=com
ou: Users
objectClass: organizationalUnit
objectClass: top
```



Create Object

Server: 172.16.128.1 Container: ou=Users,dc=datatrust,dc=com

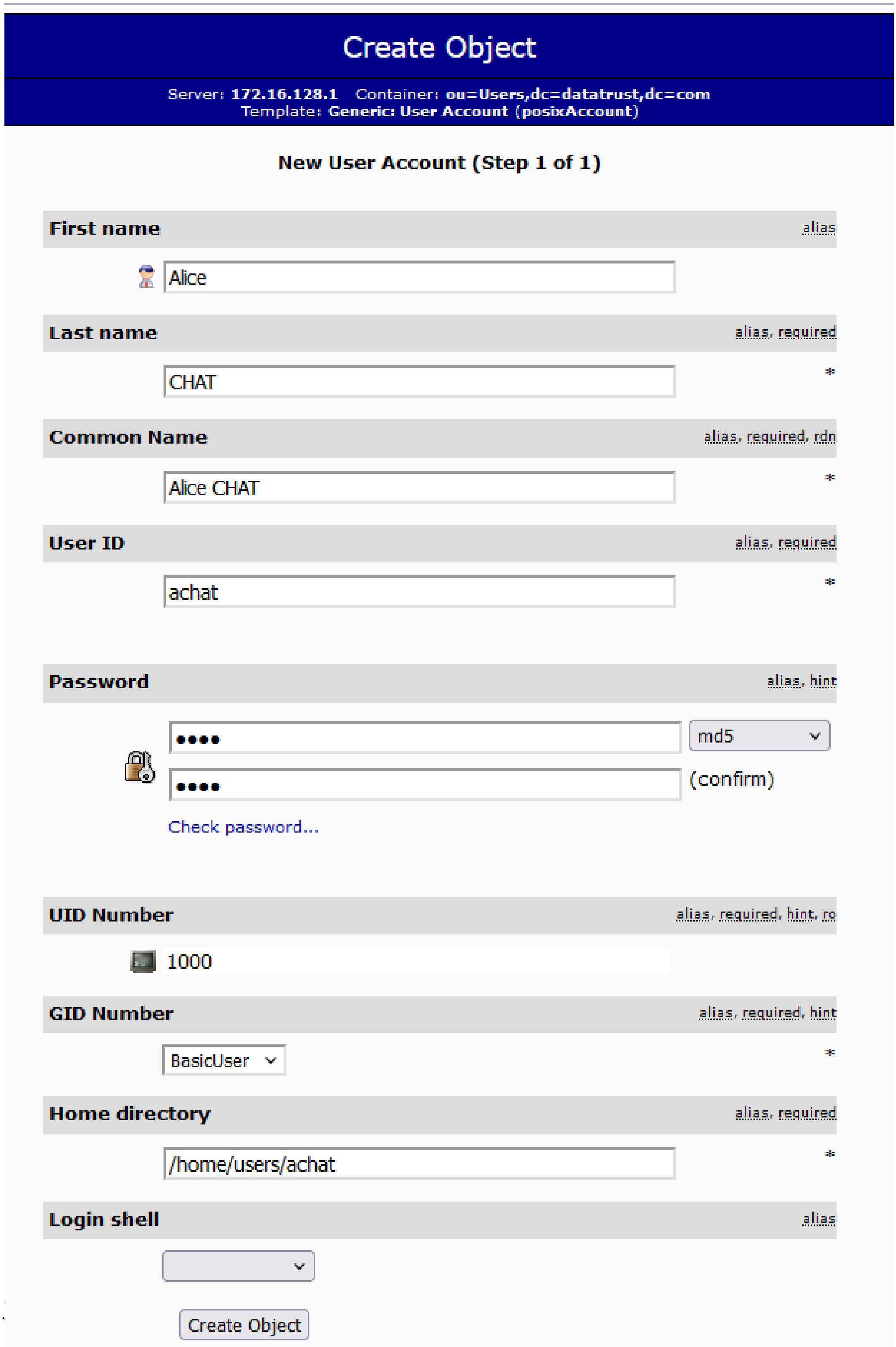
Select a template for the creation process

Templates:

○ Courier Mail: Account
○ Courier Mail: Alias
○ Generic: Address Book Entry
○ Generic: DNS Entry
○ Generic: LDAP Alias
○ Generic: Organisational Role
○ Generic: Organisational Unit
○ Generic: Posix Group
○ Generic: Simple Security Object
○ Generic: User Account
○ Kolab: User Entry
○ Samba: Account

☒ Samba: Domain
○ Samba: Group Mapping
○ Samba: Machine
☒ Sendmail: Alias
☒ Sendmail: Cluster
☒ Sendmail: Domain
☒ Sendmail: Relays
☒ Sendmail: Virtual Domain
☒ Sendmail: Virtual Users
○ Thunderbird: Address Book Entry
○ Default

Pour ALICE:



Create Object

Server: **172.16.128.1** Container: **ou=Users,dc=datatrust,dc=com**
Template: **Generic: User Account (posixAccount)**

New User Account (Step 1 of 1)

First name alias
Alice

Last name alias, required
CHAT *

Common Name alias, required, rdn
Alice CHAT *

User ID alias, required
achat *

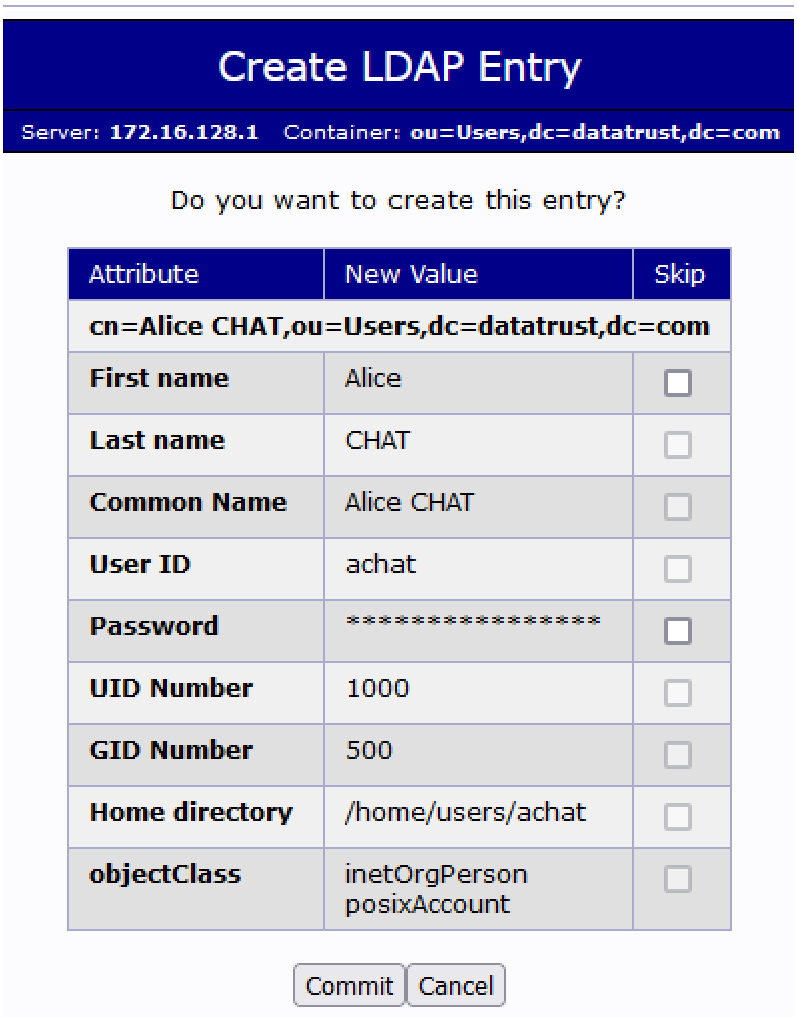
Password alias, hint
•••• md5
•••• (confirm)
Check password...

UID Number alias, required, hint, ro
1000

GID Number alias, required, hint
BasicUser *

Home directory alias, required
/home/users/achat *

Login shell alias



Create LDAP Entry

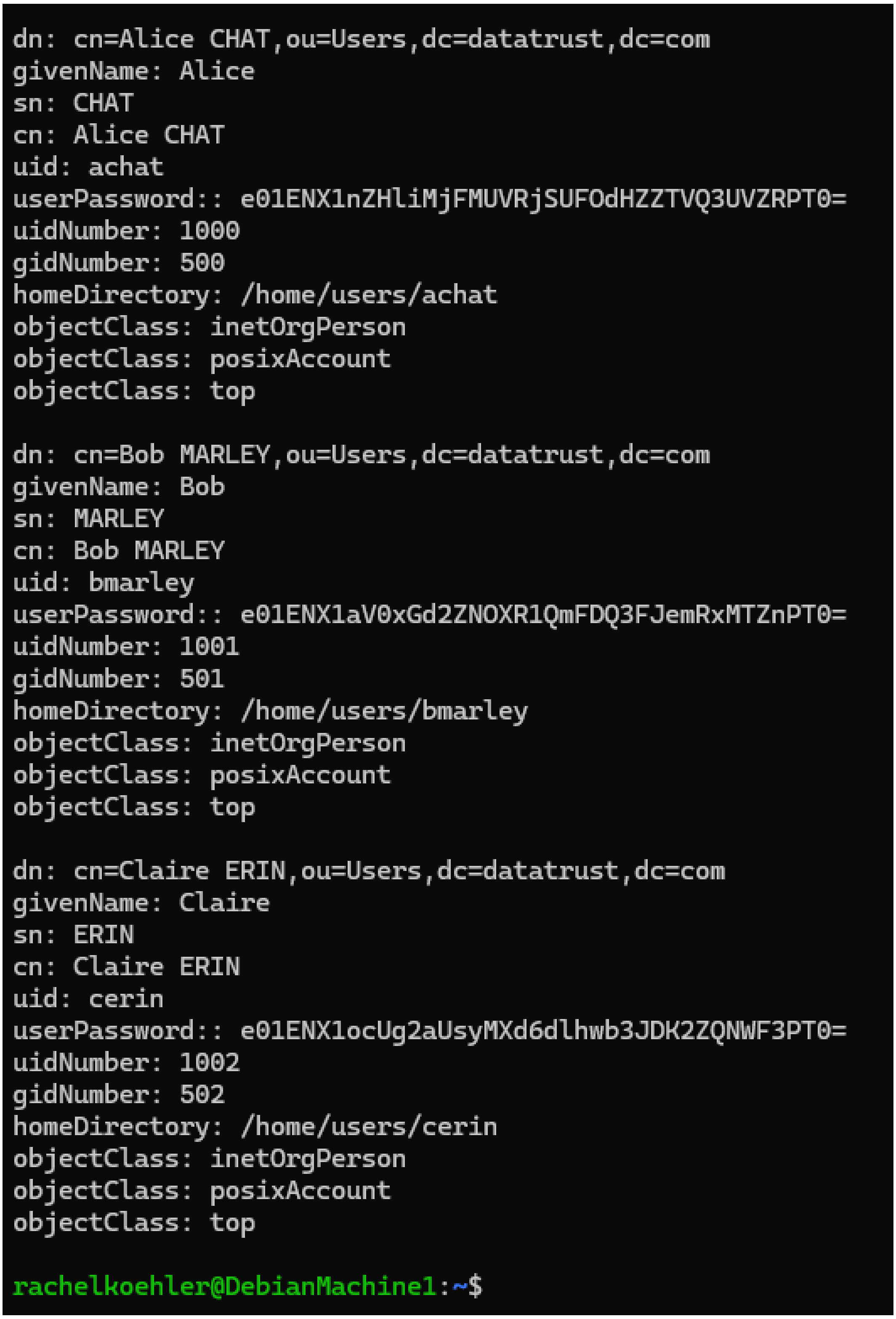
Server: **172.16.128.1** Container: **ou=Users,dc=datatrust,dc=com**

Do you want to create this entry?

Attribute	New Value	Skip
cn=Alice CHAT,ou=Users,dc=datatrust,dc=com		
First name	Alice	□
Last name	CHAT	□
Common Name	Alice CHAT	□
User ID	achat	□
Password	****************	□
UID Number	1000	□
GID Number	500	□
Home directory	/home/users/achat	□
objectClass	inetOrgPerson posixAccount	□

Pour BOB:

Pour CLAIRE:

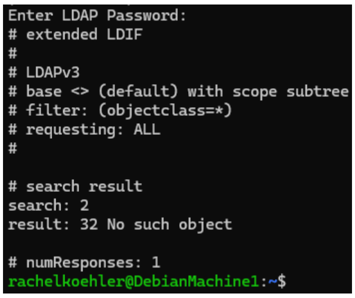


```
dn: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
givenName: Alice
sn: CHAT
cn: Alice CHAT
uid: achat
userPassword:: e01ENX1nZHliMjFMUVRjSUFOdHZZTVQ3UVZRPT0=
uidNumber: 1000
gidNumber: 500
homeDirectory: /home/users/achat
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

dn: cn=Bob MARLEY,ou=Users,dc=datatrust,dc=com
givenName: Bob
sn: MARLEY
cn: Bob MARLEY
uid: bmarley
userPassword:: e01ENX1aV0xGd2ZNOXR1QmFDQ3FJemRxMTZnPT0=
uidNumber: 1001
gidNumber: 501
homeDirectory: /home/users/bmarley
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

dn: cn=Claire ERIN,ou=Users,dc=datatrust,dc=com
givenName: Claire
sn: ERIN
cn: Claire ERIN
uid: cerin
userPassword:: e01ENX1ocUg2aUsyMXd6dlhwb3JDK2ZQNWF3PT0=
uidNumber: 1002
gidNumber: 502
homeDirectory: /home/users/cerin
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

rachelkoehler@DebianMachine1:~$
```

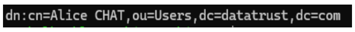
3. Test des Accès

Connectez-vous avec chaque utilisateur et vérifiez que les permissions sont appliquées correctement.

Exemple pour Alice :

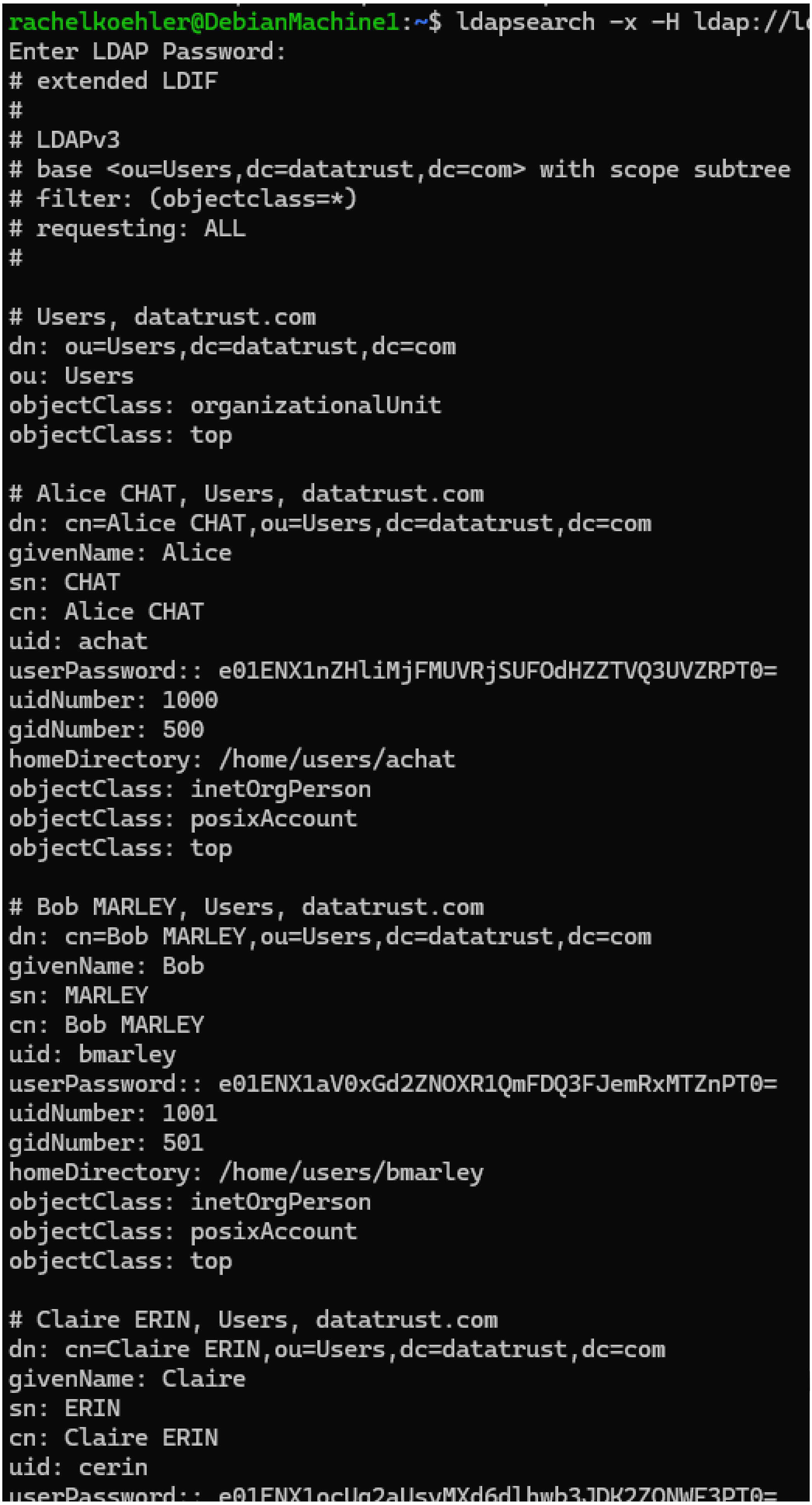
<table>
<tr><th><b><i>ALICE - info</i></b></th><th><b><i>Commande rentrée</i></b></th><th><b><i>Erreur rencontrée</i></b></th></tr>
<tr><td>dn: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
givenName: Alice
sn: CHAT
cn: Alice CHAT
uid: achat
userPassword:: e01ENX1nZHliMjFMUVRjSUFOdHZZTVQ3UVZRPT0=
uidNumber: 1000
gidNumber: 500
homeDirectory: /home/users/achat
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top</td><td>ldapsearch -x -H ldap://localhost -D "cn=Alice CHAT,ou=Users ,dc=datatrust,dc=com" -W</td><td>

```
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <> (default) with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# search result
search: 2
result: 32 No such object

# numResponses: 1
rachelkoehler@DebianMachine1:~$
```

<b><i>This means that LDAP cannot find the specified</i></b></td></tr>
<tr><td></td><td><i>ldapwhoami -x -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com" -W</i></td><td>

```
dn:cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
```

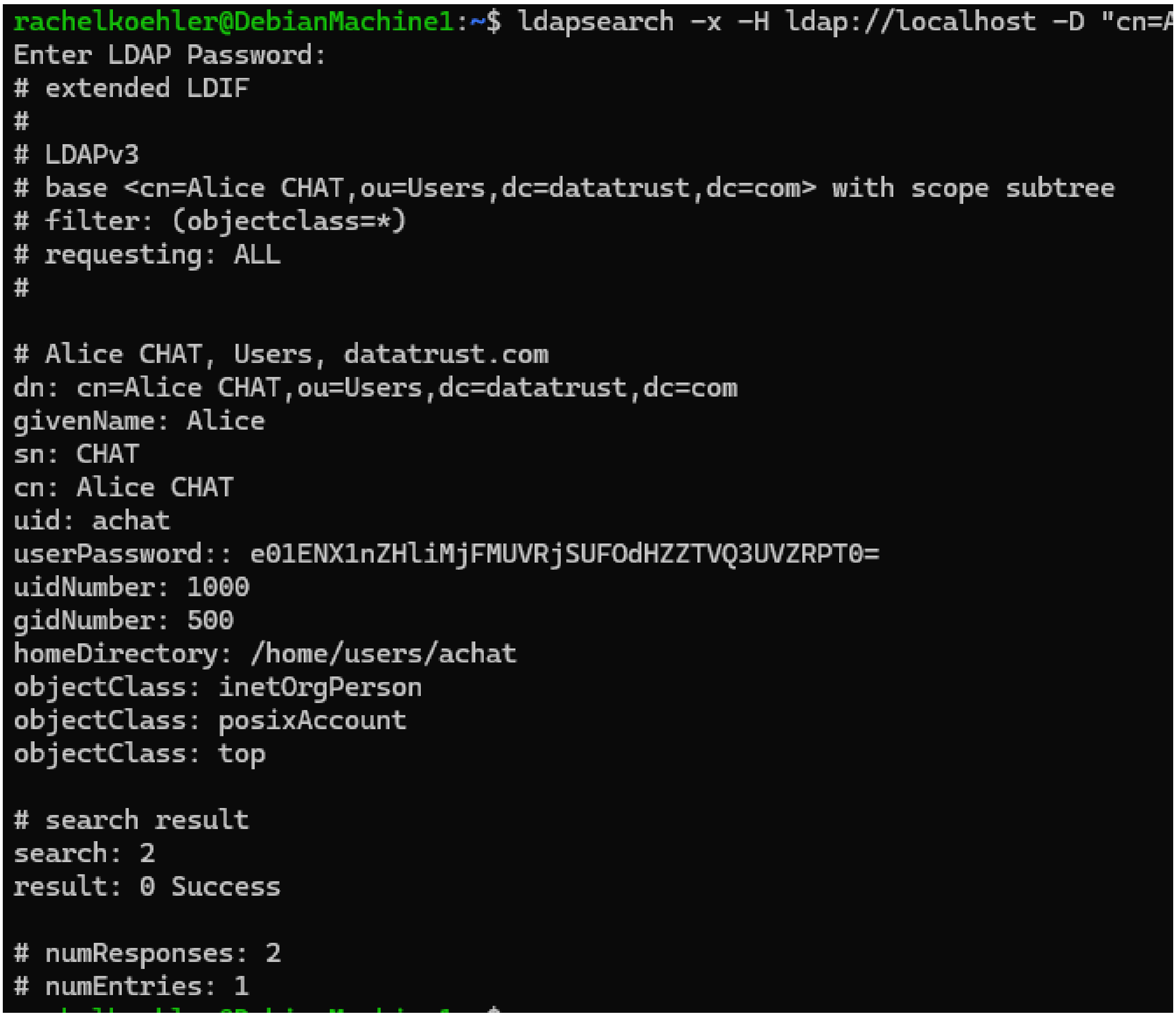
<b><i>This means that it works</i></b></td></tr>
<tr><td colspan="3"><b><i>She may need some access/permissions car la command marche pour l’Admin:</i></b>
<i>ldapsearch -x -H ldap://localhost -D "cn=admin,dc=datatrust,dc=com" -W -b "ou=Users,dc=datatrust,dc=com"</i></td></tr>
</table>



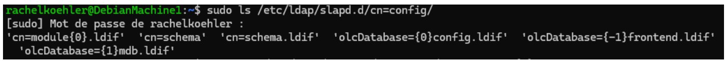
```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://l
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <ou=Users,dc=datatrust,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# Users, datatrust.com
dn: ou=Users,dc=datatrust,dc=com
ou: Users
objectClass: organizationalUnit
objectClass: top

# Alice CHAT, Users, datatrust.com
dn: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
givenName: Alice
sn: CHAT
cn: Alice CHAT
uid: achat
userPassword:: e01ENX1nZHliMjFMUVRjSUFOdHZZTVQ3UVZRPT0=
uidNumber: 1000
gidNumber: 500
homeDirectory: /home/users/achat
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# Bob MARLEY, Users, datatrust.com
dn: cn=Bob MARLEY,ou=Users,dc=datatrust,dc=com
givenName: Bob
sn: MARLEY
cn: Bob MARLEY
uid: bmarley
userPassword:: e01ENX1aV0xGd2ZNOXR1QmFDQ3FJemRxMTZnPT0=
uidNumber: 1001
gidNumber: 501
homeDirectory: /home/users/bmarley
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# Claire ERIN, Users, datatrust.com
dn: cn=Claire ERIN,ou=Users,dc=datatrust,dc=com
givenName: Claire
sn: ERIN
cn: Claire ERIN
uid: cerin
userPassword:: e01ENX1ocUg2aUsvMXd6dlhwb3JDK2ZONWE3PT0=
```

Aussi elle est capable de lire son propre fichier ldif, via la commande
ldapsearch -x -H ldap://localhost -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com" -W -b "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com"

Donc c’est définitivement un pb de permissions

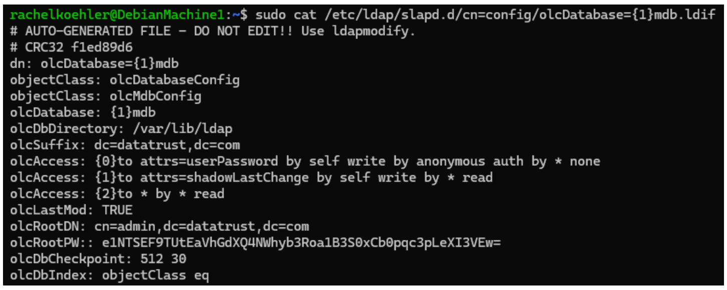


```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://localhost -D "cn=A
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <cn=Alice CHAT,ou=Users,dc=datatrust,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# Alice CHAT, Users, datatrust.com
dn: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
givenName: Alice
sn: CHAT
cn: Alice CHAT
uid: achat
userPassword:: e01ENX1nZHliMjFMUVRjSUFOdHZZTVQ3UVZRPT0=
uidNumber: 1000
gidNumber: 500
homeDirectory: /home/users/achat
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```



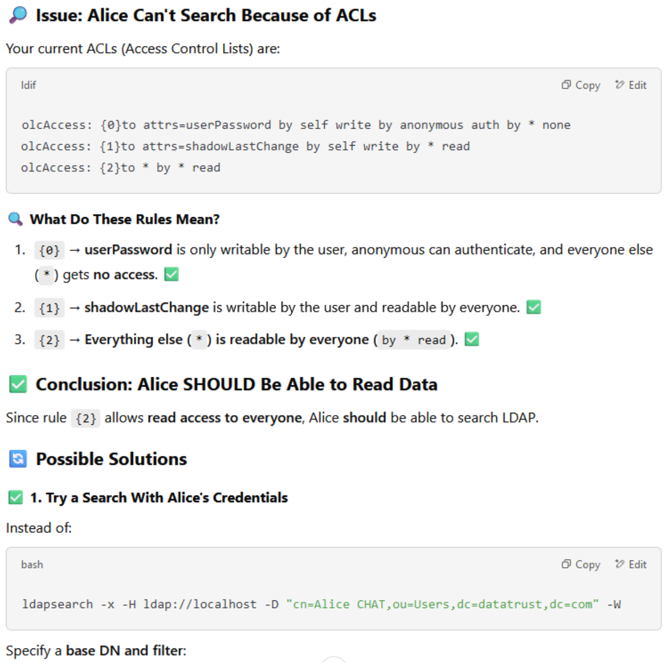
```
rachelkoehler@DebianMachine1:~$ sudo ls /etc/ldap/slapd.d/cn=config/
[sudo] Mot de passe de rachelkoehler :
'cn=module{0}.ldif'  'cn=schema'  'cn=schema.ldif'  'olcDatabase={0}config.ldif'  'olcDatabase={-1}frontend.ldif'
  'olcDatabase={1}mdb.ldif'
```

Go in the mdb.ldif file



```
rachelkoehler@DebianMachine1:~$ sudo cat /etc/ldap/slapd.d/cn=config/olcDatabase={1}mdb.ldif
# AUTO-GENERATED FILE - DO NOT EDIT!! Use ldapmodify.
# CRC32 f1ed89d6
dn: olcDatabase={1}mdb
objectClass: olcDatabaseConfig
objectClass: olcMdbConfig
olcDatabase: {1}mdb
olcDbDirectory: /var/lib/ldap
olcSuffix: dc=datatrust,dc=com
olcAccess: {0}to attrs=userPassword by self write by anonymous auth by * none
olcAccess: {1}to attrs=shadowLastChange by self write by * read
olcAccess: {2}to * by * read
olcLastMod: TRUE
olcRootDN: cn=admin,dc=datatrust,dc=com
olcRootPW:: e1NTSEF9TUtEaVhGdXQ4NWhyb3RoalB3S0xCb0pqc3pLeXI3VEw=
olcDbCheckpoint: 512 30
olcDbIndex: objectClass eq
```

SOLUTION



🔎 Issue: Alice Can't Search Because of ACLs

Your current ACLs (Access Control Lists) are:

```ldif
olcAccess: {0}to attrs=userPassword by self write by anonymous auth by * none
olcAccess: {1}to attrs=shadowLastChange by self write by * read
olcAccess: {2}to * by * read
```

🔍 What Do These Rules Mean?

1. `{0}` → **userPassword** is only writable by the user, anonymous can authenticate, and everyone else (`*`) gets **no access**. ✅
2. `{1}` → **shadowLastChange** is writable by the user and readable by everyone. ✅
3. `{2}` → **Everything else** (`*`) **is readable by everyone** (`by * read`). ✅

✅ Conclusion: Alice SHOULD Be Able to Read Data

Since rule `{2}` allows **read access to everyone**, Alice **should** be able to search LDAP.

🔄 Possible Solutions

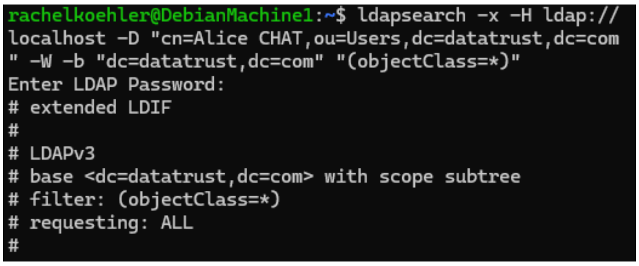
✅ 1. Try a Search With Alice's Credentials

Instead of:

```bash
ldapsearch -x -H ldap://localhost -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com" -W
```

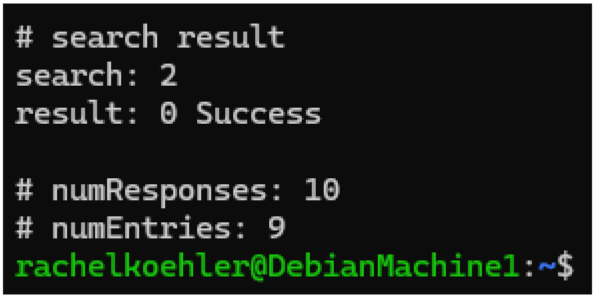
Specify a **base DN and filter**:

ldapsearch -x -H ldap://localhost -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com" -W -b "dc=datatrust,dc=com" "(objectClass=*)"



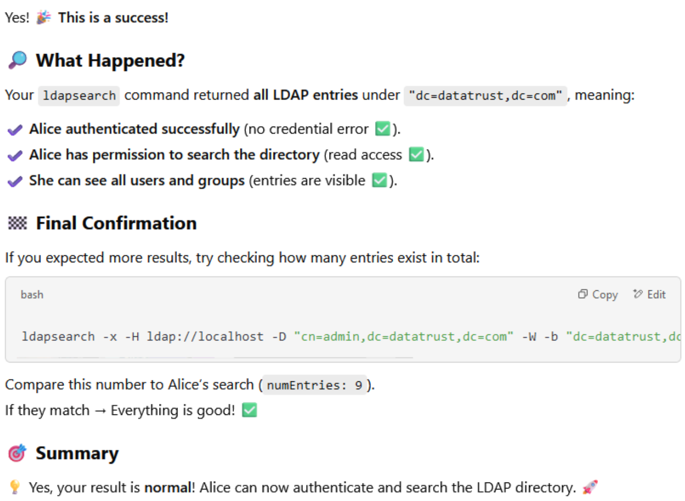
```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://
localhost -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
" -W -b "dc=datatrust,dc=com" "(objectClass=*)"
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=datatrust,dc=com> with scope subtree
# filter: (objectClass=*)
# requesting: ALL
#
```

(...)



```
# search result
search: 2
result: 0 Success

# numResponses: 10
# numEntries: 9
rachelkoehler@DebianMachine1:~$
```

SUCCESS !!!!



Yes! 🎉 **This is a success!**

🔎 What Happened?

Your `ldapsearch` command returned **all LDAP entries** under `"dc=datatrust,dc=com"`, meaning:

- ✔ **Alice authenticated successfully** (no credential error ✅).
- ✔ **Alice has permission to search the directory** (read access ✅).
- ✔ **She can see all users and groups** (entries are visible ✅).

🏁 Final Confirmation

If you expected more results, try checking how many entries exist in total:

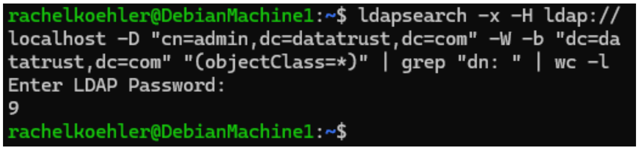
```bash
ldapsearch -x -H ldap://localhost -D "cn=admin,dc=datatrust,dc=com" -W -b "dc=datatrust,dc
```

Compare this number to Alice's search (`numEntries: 9`).
If they match → Everything is good! ✅

🎯 Summary

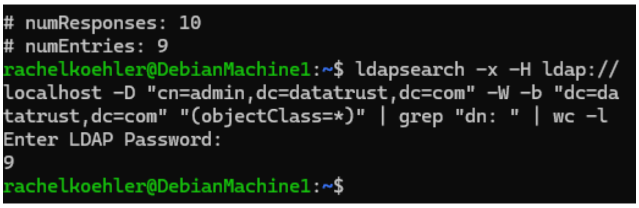
💡 Yes, your result is **normal**! Alice can now authenticate and search the LDAP directory. 🚀

Compare to Alice search (from screen above):

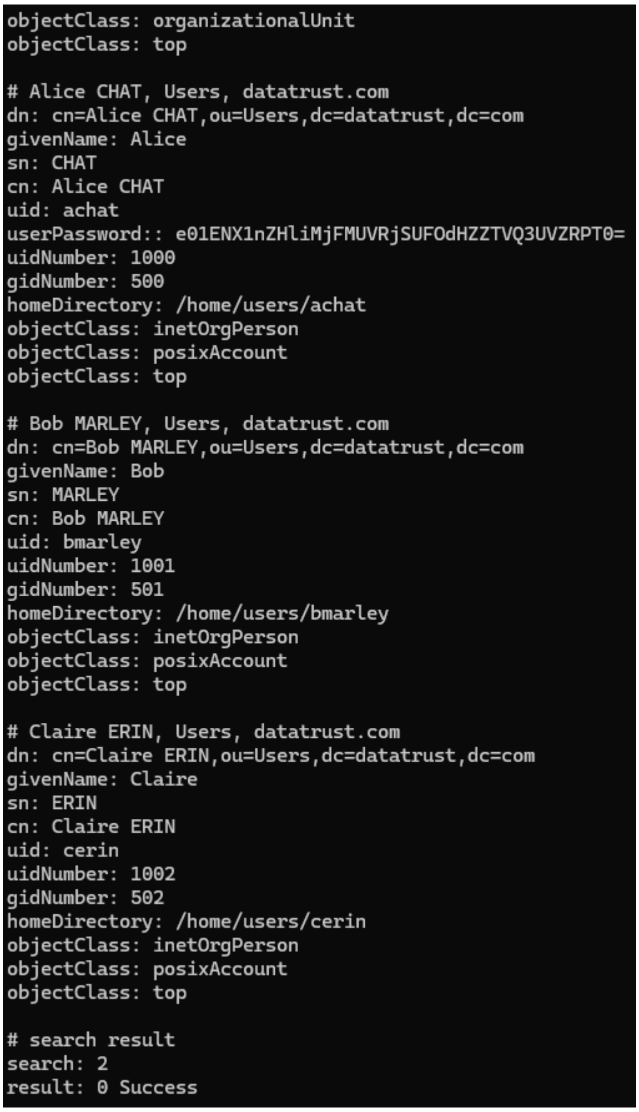


```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://
localhost -D "cn=admin,dc=datatrust,dc=com" -W -b "dc=da
tatrust,dc=com" "(objectClass=*)" | grep "dn: " | wc -l
Enter LDAP Password:
9
rachelkoehler@DebianMachine1:~$
```

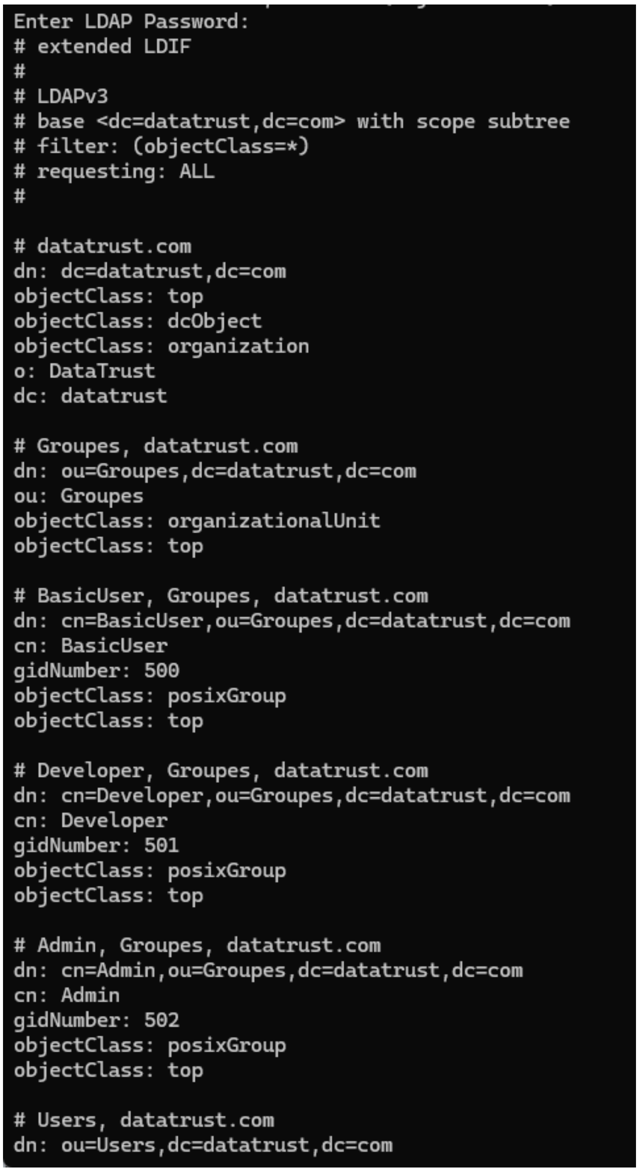
or (well it's the same)



```
# numResponses: 10
# numEntries: 9
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://
localhost -D "cn=admin,dc=datatrust,dc=com" -W -b "dc=da
tatrust,dc=com" "(objectClass=*)" | grep "dn: " | wc -l
Enter LDAP Password:
9
rachelkoehler@DebianMachine1:~$
```

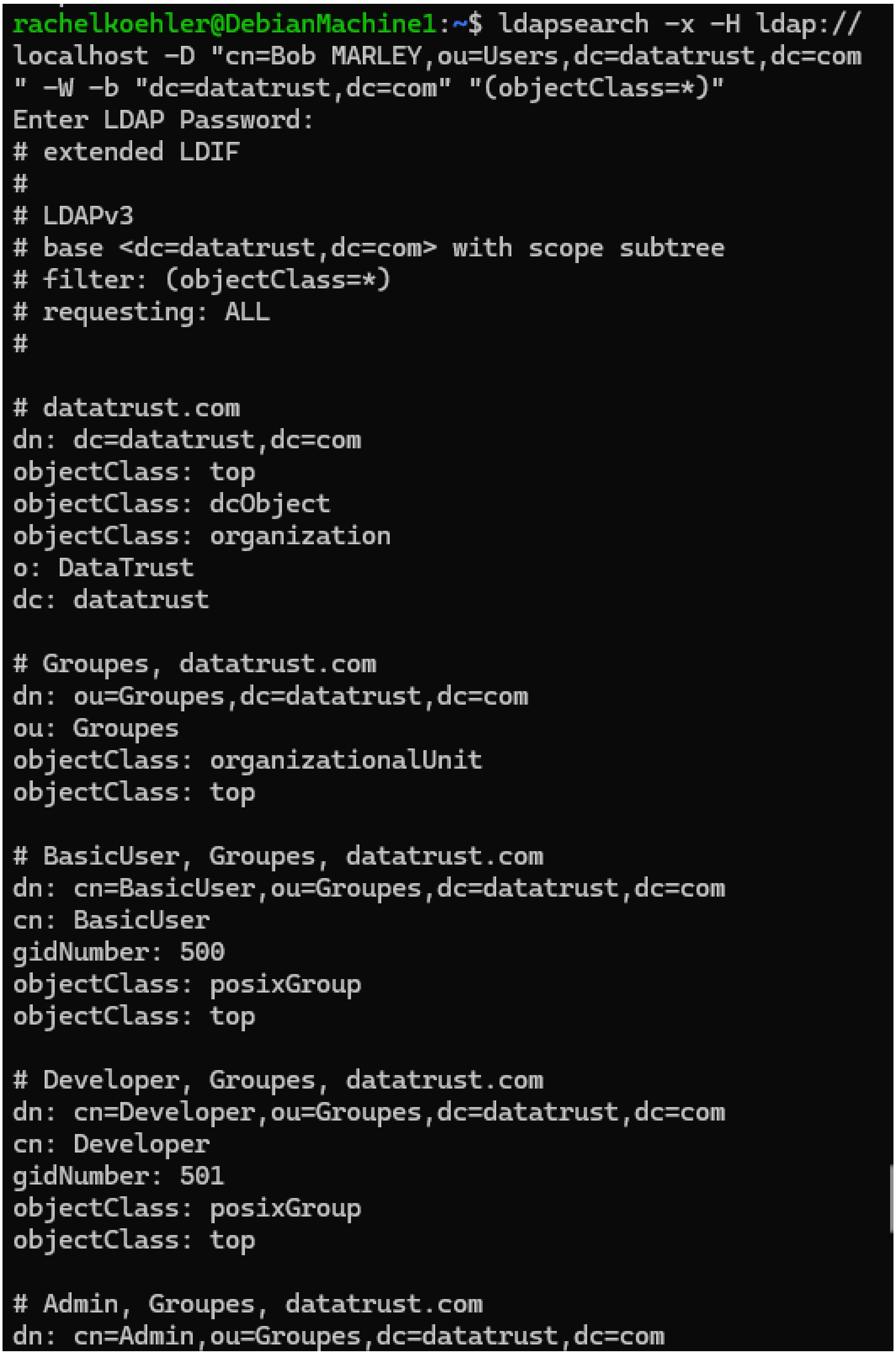


```
objectClass: organizationalUnit
objectClass: top

# Alice CHAT, Users, datatrust.com
dn: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
givenName: Alice
sn: CHAT
cn: Alice CHAT
uid: achat
userPassword:: e01ENX1nZHliMjFMUVRjSUFOdHZZTVQ3UVZRPT0=
uidNumber: 1000
gidNumber: 500
homeDirectory: /home/users/achat
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# Bob MARLEY, Users, datatrust.com
dn: cn=Bob MARLEY,ou=Users,dc=datatrust,dc=com
givenName: Bob
sn: MARLEY
cn: Bob MARLEY
uid: bmarley
uidNumber: 1001
gidNumber: 501
homeDirectory: /home/users/bmarley
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# Claire ERIN, Users, datatrust.com
dn: cn=Claire ERIN,ou=Users,dc=datatrust,dc=com
givenName: Claire
sn: ERIN
cn: Claire ERIN
uid: cerin
uidNumber: 1002
gidNumber: 502
homeDirectory: /home/users/cerin
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# search result
search: 2
result: 0 Success
```

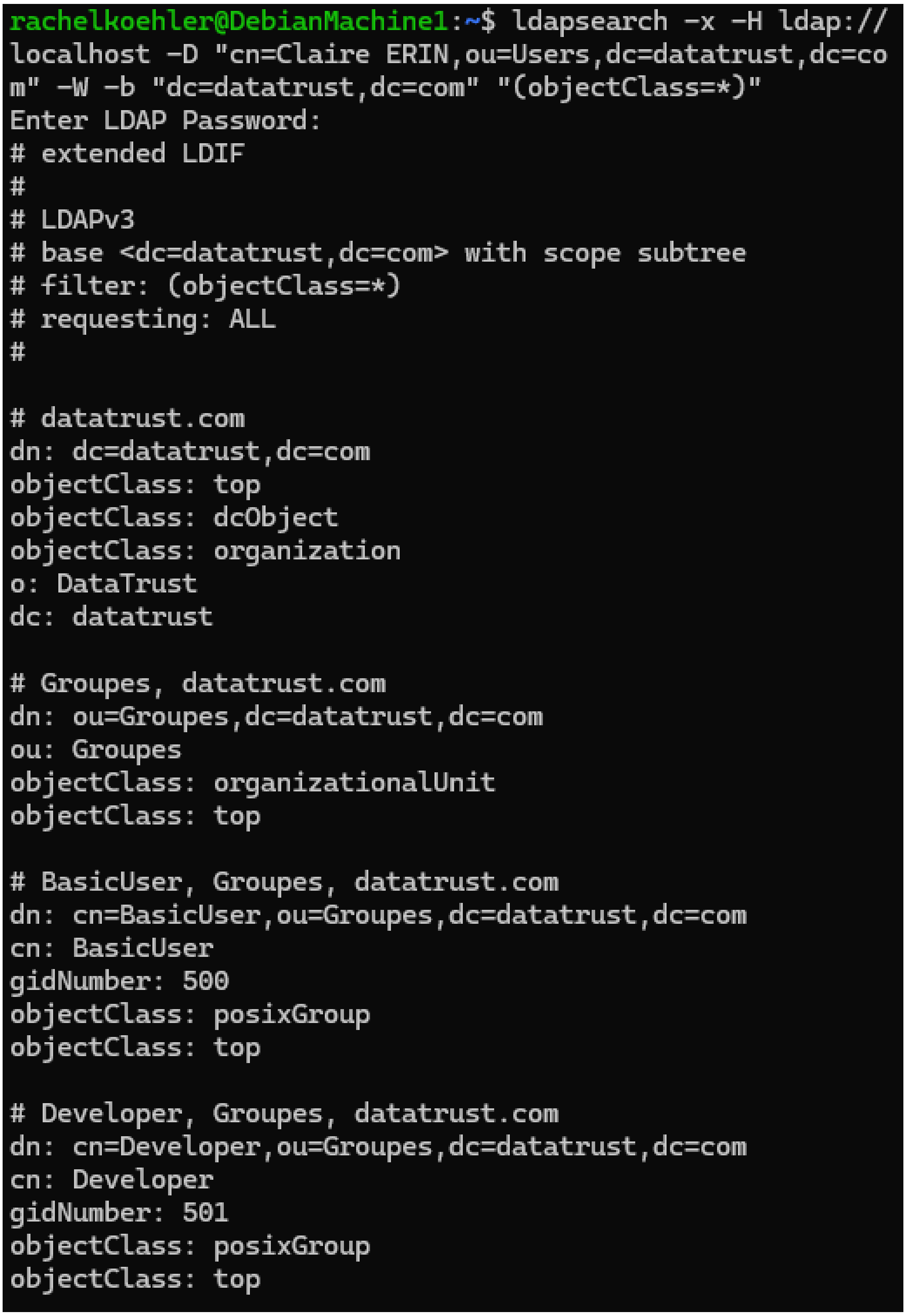
```
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=datatrust,dc=com> with scope subtree
# filter: (objectClass=*)
# requesting: ALL
#

# datatrust.com
dn: dc=datatrust,dc=com
objectClass: top
objectClass: dcObject
objectClass: organization
o: DataTrust
dc: datatrust

# Groupes, datatrust.com
dn: ou=Groupes,dc=datatrust,dc=com
ou: Groupes
objectClass: organizationalUnit
objectClass: top

# BasicUser, Groupes, datatrust.com
dn: cn=BasicUser,ou=Groupes,dc=datatrust,dc=com
cn: BasicUser
gidNumber: 500
objectClass: posixGroup
objectClass: top

# Developer, Groupes, datatrust.com
dn: cn=Developer,ou=Groupes,dc=datatrust,dc=com
cn: Developer
gidNumber: 501
objectClass: posixGroup
objectClass: top

# Admin, Groupes, datatrust.com
dn: cn=Admin,ou=Groupes,dc=datatrust,dc=com
cn: Admin
gidNumber: 502
objectClass: posixGroup
objectClass: top

# Users, datatrust.com
dn: ou=Users,dc=datatrust,dc=com
```

Fonctionne pour Bob aussi:



```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://
localhost -D "cn=Bob MARLEY,ou=Users,dc=datatrust,dc=com
" -W -b "dc=datatrust,dc=com" "(objectClass=*)"
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=datatrust,dc=com> with scope subtree
# filter: (objectClass=*)
# requesting: ALL
#

# datatrust.com
dn: dc=datatrust,dc=com
objectClass: top
objectClass: dcObject
objectClass: organization
o: DataTrust
dc: datatrust

# Groupes, datatrust.com
dn: ou=Groupes,dc=datatrust,dc=com
ou: Groupes
objectClass: organizationalUnit
objectClass: top

# BasicUser, Groupes, datatrust.com
dn: cn=BasicUser,ou=Groupes,dc=datatrust,dc=com
cn: BasicUser
gidNumber: 500
objectClass: posixGroup
objectClass: top

# Developer, Groupes, datatrust.com
dn: cn=Developer,ou=Groupes,dc=datatrust,dc=com
cn: Developer
gidNumber: 501
objectClass: posixGroup
objectClass: top

# Admin, Groupes, datatrust.com
dn: cn=Admin,ou=Groupes,dc=datatrust,dc=com
```

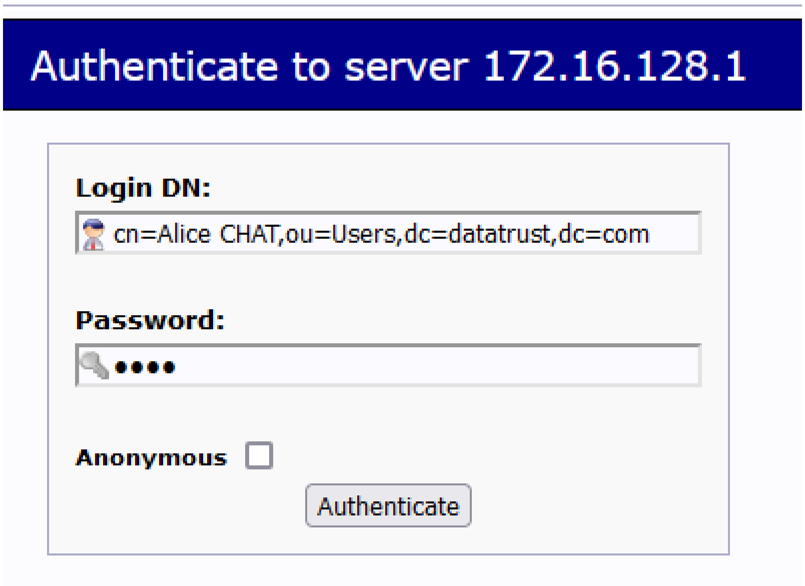
Idem pour Claire:



```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://
localhost -D "cn=Claire ERIN,ou=Users,dc=datatrust,dc=co
m" -W -b "dc=datatrust,dc=com" "(objectClass=*)"
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=datatrust,dc=com> with scope subtree
# filter: (objectClass=*)
# requesting: ALL
#

# datatrust.com
dn: dc=datatrust,dc=com
objectClass: top
objectClass: dcObject
objectClass: organization
o: DataTrust
dc: datatrust

# Groupes, datatrust.com
dn: ou=Groupes,dc=datatrust,dc=com
ou: Groupes
objectClass: organizationalUnit
objectClass: top

# BasicUser, Groupes, datatrust.com
dn: cn=BasicUser,ou=Groupes,dc=datatrust,dc=com
cn: BasicUser
gidNumber: 500
objectClass: posixGroup
objectClass: top

# Developer, Groupes, datatrust.com
dn: cn=Developer,ou=Groupes,dc=datatrust,dc=com
cn: Developer
gidNumber: 501
objectClass: posixGroup
objectClass: top
```

AVEC **PHPLDAPADMIN**

Jean-François (Jeff) **ANDRIATSIMEVA** - Moussa **CAMARA** Rachel **KOEHLER** - Thomas **VERAN**

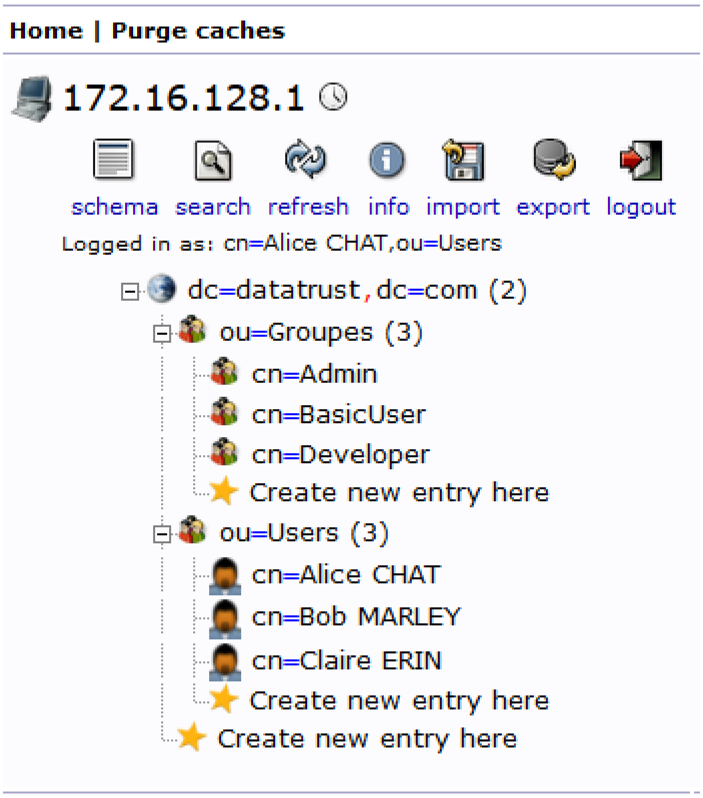


Authenticate to server 172.16.128.1

Login DN: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com

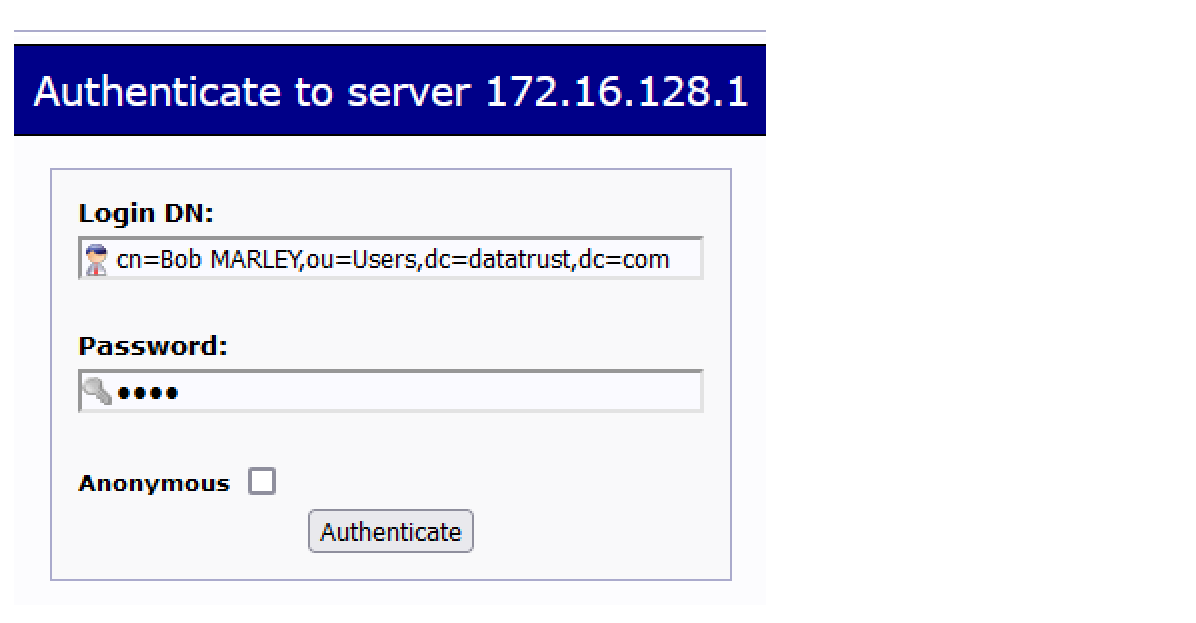
Password: ••••

Anonymous □



Logged in as: cn=Alice CHAT,ou=Users

```
⊟ dc=datatrust,dc=com (2)
  ⊟ ou=Groupes (3)
      cn=Admin
      cn=BasicUser
      cn=Developer
      Create new entry here
  ⊟ ou=Users (3)
      cn=Alice CHAT
      cn=Bob MARLEY
      cn=Claire ERIN
      Create new entry here
    Create new entry here
```



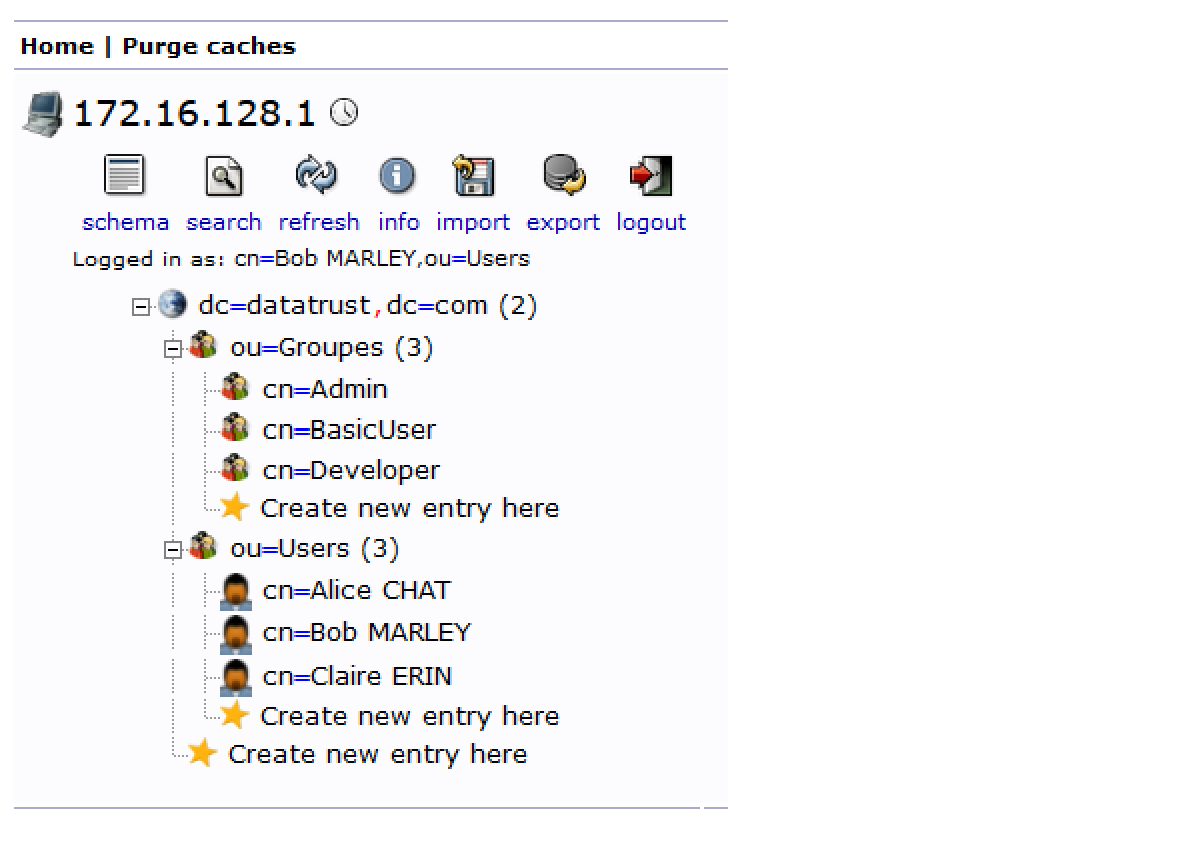
Authenticate to server 172.16.128.1

Login DN:
cn=Bob MARLEY,ou=Users,dc=datatrust,dc=com

Password:
••••

Anonymous □

Authenticate

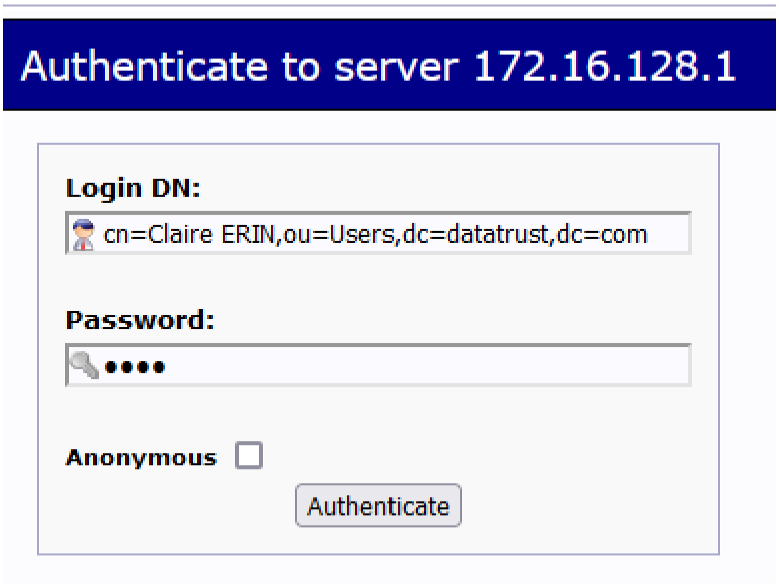


Home | Purge caches

172.16.128.1

Logged in as: cn=Bob MARLEY,ou=Users

```
dc=datatrust,dc=com (2)
  ou=Groupes (3)
    cn=Admin
    cn=BasicUser
    cn=Developer
    Create new entry here
  ou=Users (3)
    cn=Alice CHAT
    cn=Bob MARLEY
    cn=Claire ERIN
    Create new entry here
  Create new entry here
```



Authenticate to server 172.16.128.1

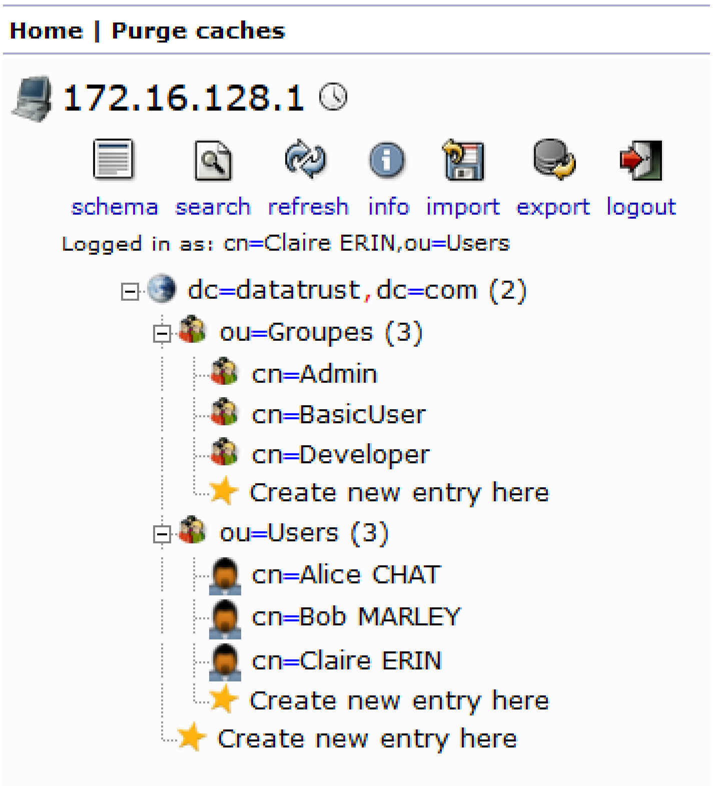
Login DN:

cn=Claire ERIN,ou=Users,dc=datatrust,dc=com

Password:

••••

Anonymous □



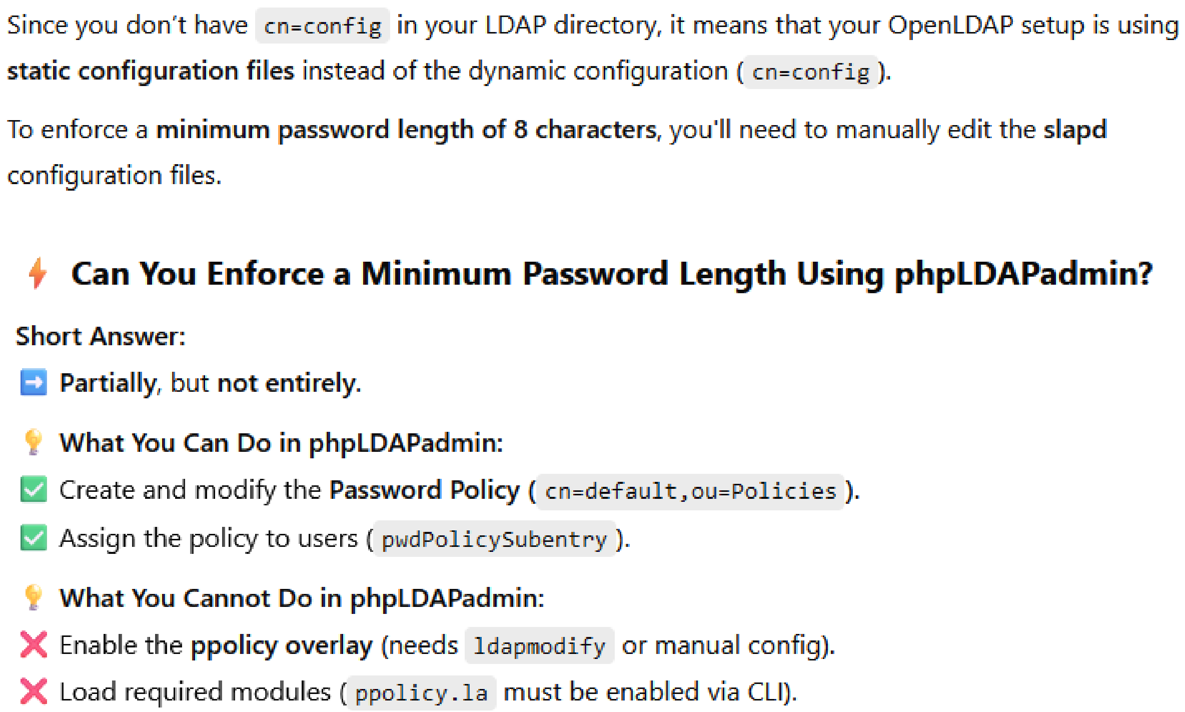
Logged in as: cn=Claire ERIN,ou=Users

```
⊟ dc=datatrust,dc=com (2)
    ⊟ ou=Groupes (3)
        cn=Admin
        cn=BasicUser
        cn=Developer
        Create new entry here
    ⊟ ou=Users (3)
        cn=Alice CHAT
        cn=Bob MARLEY
        cn=Claire ERIN
        Create new entry here
    Create new entry here
```

Job 3 : Options de sécurité de base

1. Gestion des Mots de Passe

Configurez LDAP pour qu'il impose une longueur minimale de mot de passe (au moins 8 caractères) et documentez la configuration.
Comme je suis en mode statique je n'ai pas la possibilité de passer par "cn=config"



Since you don't have `cn=config` in your LDAP directory, it means that your OpenLDAP setup is using **static configuration files** instead of the dynamic configuration (`cn=config`).

To enforce a **minimum password length of 8 characters**, you'll need to manually edit the **slapd** configuration files.

⚡ Can You Enforce a Minimum Password Length Using phpLDAPadmin?

Short Answer:

➡️ **Partially**, but **not entirely**.

💡 **What You Can Do in phpLDAPadmin:**

✅ Create and modify the **Password Policy** (`cn=default,ou=Policies`).

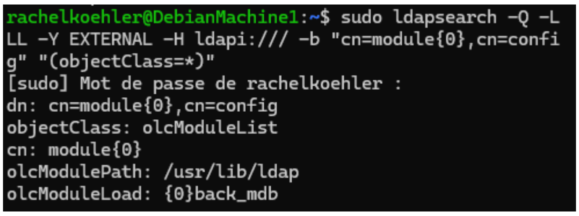
✅ Assign the policy to users (`pwdPolicySubentry`).

💡 **What You Cannot Do in phpLDAPadmin:**

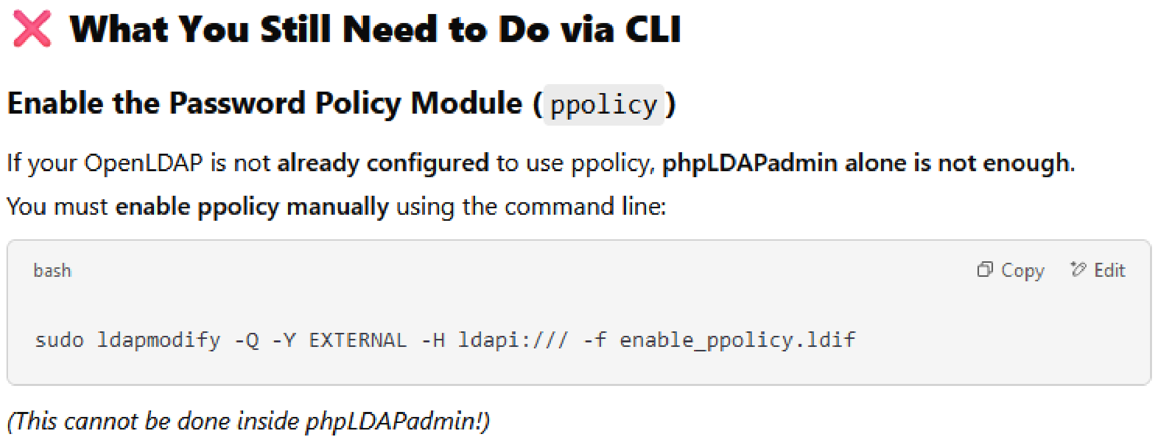
❌ Enable the **ppolicy overlay** (needs `ldapmodify` or manual config).

❌ Load required modules (`ppolicy.la` must be enabled via CLI).

Vérification existence fichier ppolicy


`rachelkoehler@DebianMachine1:~$ sudo ldapsearch -Q -LLL -Y EXTERNAL -H ldapi:/// -b "cn=module{0},cn=config" "(objectClass=*)"`
`[sudo] Mot de passe de rachelkoehler :`
`dn: cn=module{0},cn=config`
`objectClass: olcModuleList`
`cn: module{0}`
`olcModulePath: /usr/lib/ldap`
`olcModuleLoad: {0}back_mdb`
	On remarque l'absence de "ppolicy"

Nécessite d'activer le module ppolicy pour arriver au bout du processus de gestion des mots de passe, donc va devoir passer par la CLI par la suite:



❌ **What You Still Need to Do via CLI**

Enable the Password Policy Module (`ppolicy`)

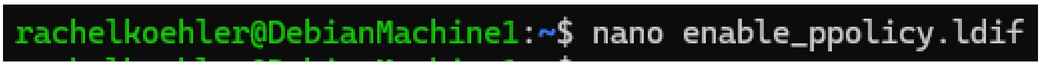
If your OpenLDAP is not **already configured** to use ppolicy, **phpLDAPadmin alone is not enough.**

You must **enable ppolicy manually** using the command line:

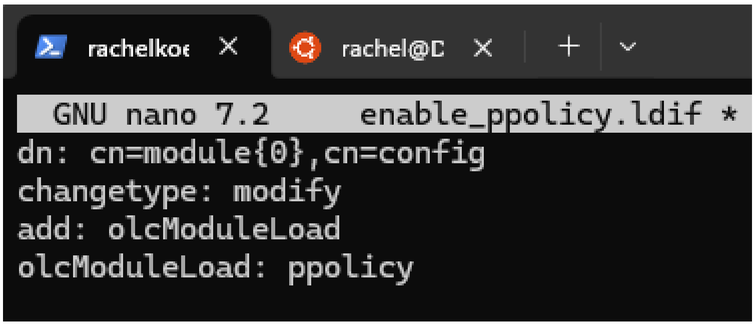
```bash
sudo ldapmodify -Q -Y EXTERNAL -H ldapi:/// -f enable_ppolicy.ldif
```

(This cannot be done inside phpLDAPadmin!)

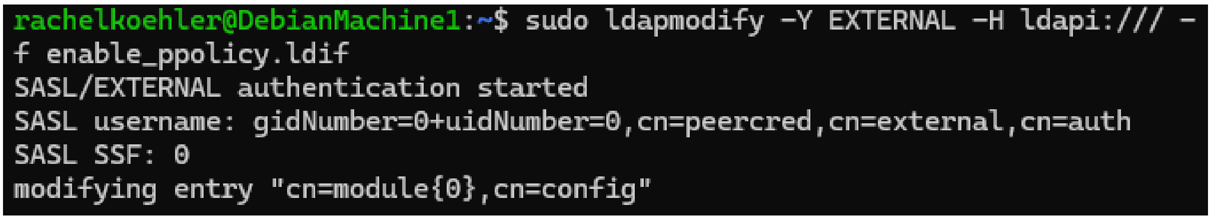
Création du fichier enable_ppolicy.ldif



```
rachelkoehler@DebianMachine1:~$ nano enable_ppolicy.ldif
```



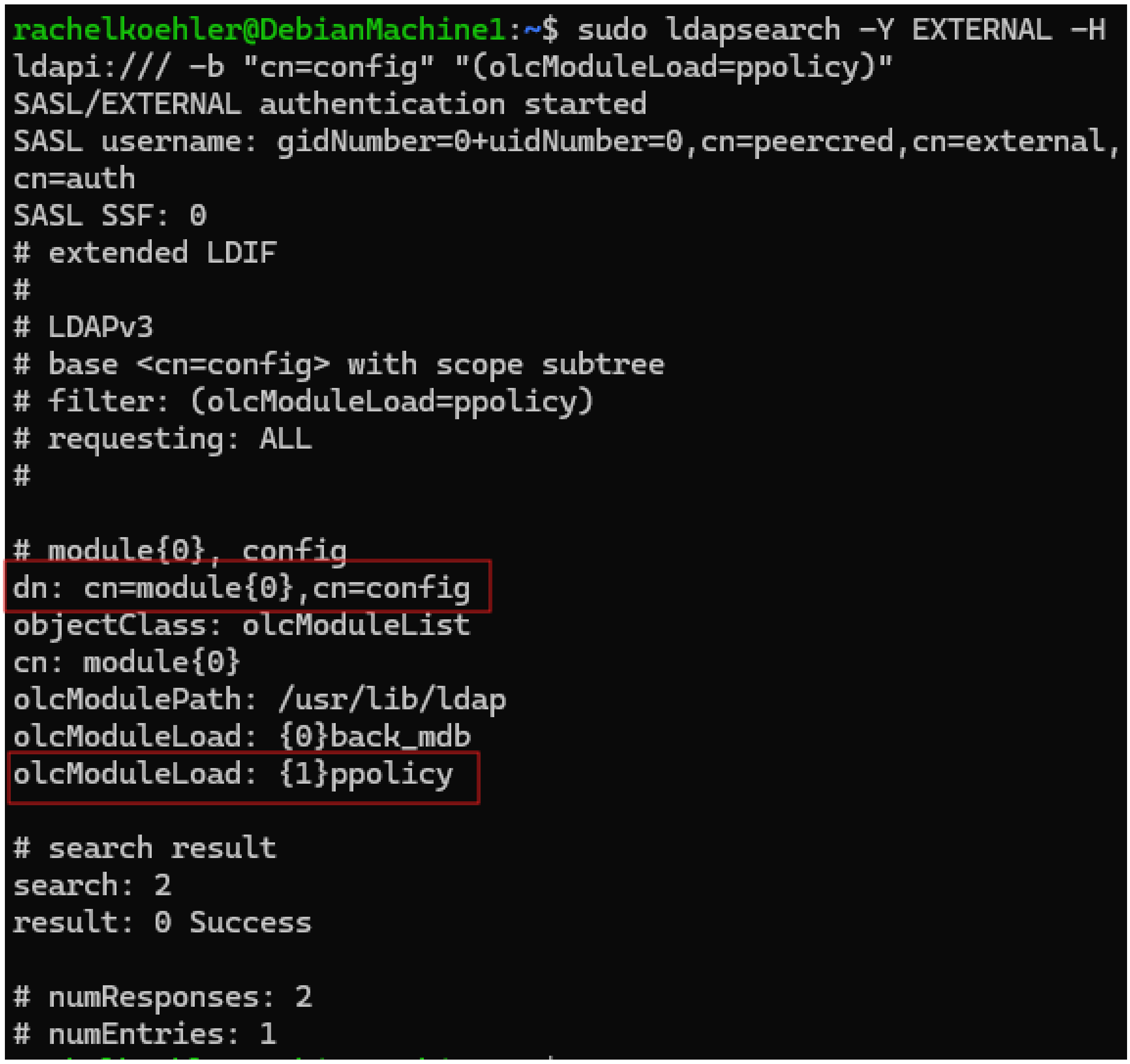
```
  GNU nano 7.2        enable_ppolicy.ldif *
dn: cn=module{0},cn=config
changetype: modify
add: olcModuleLoad
olcModuleLoad: ppolicy
```



```
rachelkoehler@DebianMachine1:~$ sudo ldapmodify -Y EXTERNAL -H ldapi:/// -
f enable_ppolicy.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
modifying entry "cn=module{0},cn=config"
```

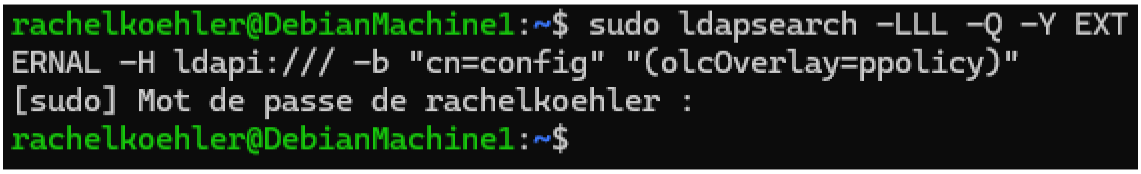
Le message “modifying entry…” nous indique bien que le fichier .ldif est bien créé…

Vérifier que ppolicy soit bien loadé



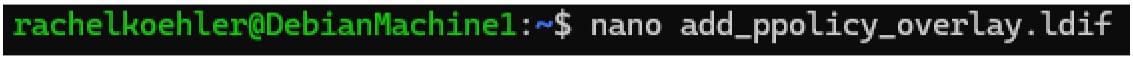
```
rachelkoehler@DebianMachine1:~$ sudo ldapsearch -Y EXTERNAL -H
ldapi:/// -b "cn=config" "(olcModuleLoad=ppolicy)"
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,
cn=auth
SASL SSF: 0
# extended LDIF
#
# LDAPv3
# base <cn=config> with scope subtree
# filter: (olcModuleLoad=ppolicy)
# requesting: ALL
#

# module{0}, config
dn: cn=module{0},cn=config
objectClass: olcModuleList
cn: module{0}
olcModulePath: /usr/lib/ldap
olcModuleLoad: {0}back_mdb
olcModuleLoad: {1}ppolicy

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

Vérifier qu'il soit "enabled":

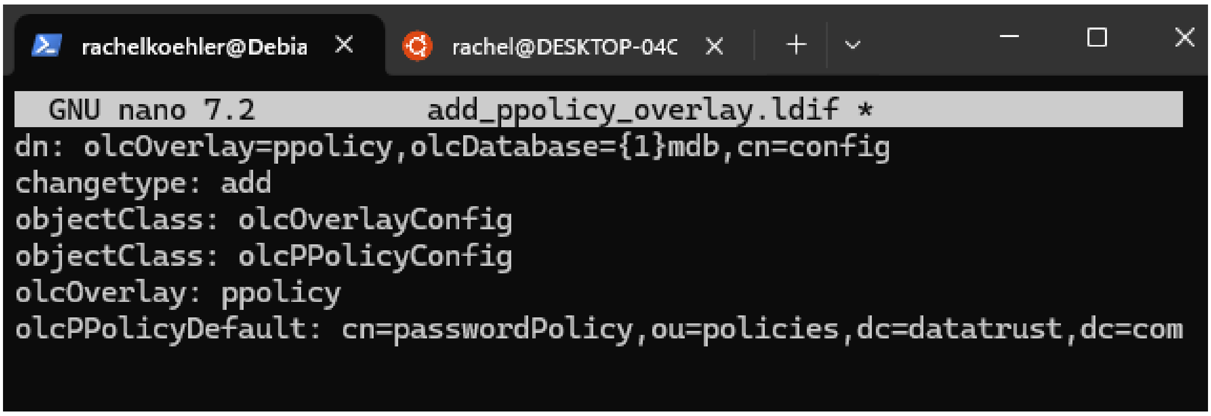


```
rachelkoehler@DebianMachine1:~$ sudo ldapsearch -LLL -Q -Y EXT
ERNAL -H ldapi:/// -b "cn=config" "(olcOverlay=ppolicy)"
[sudo] Mot de passe de rachelkoehler :
rachelkoehler@DebianMachine1:~$
```

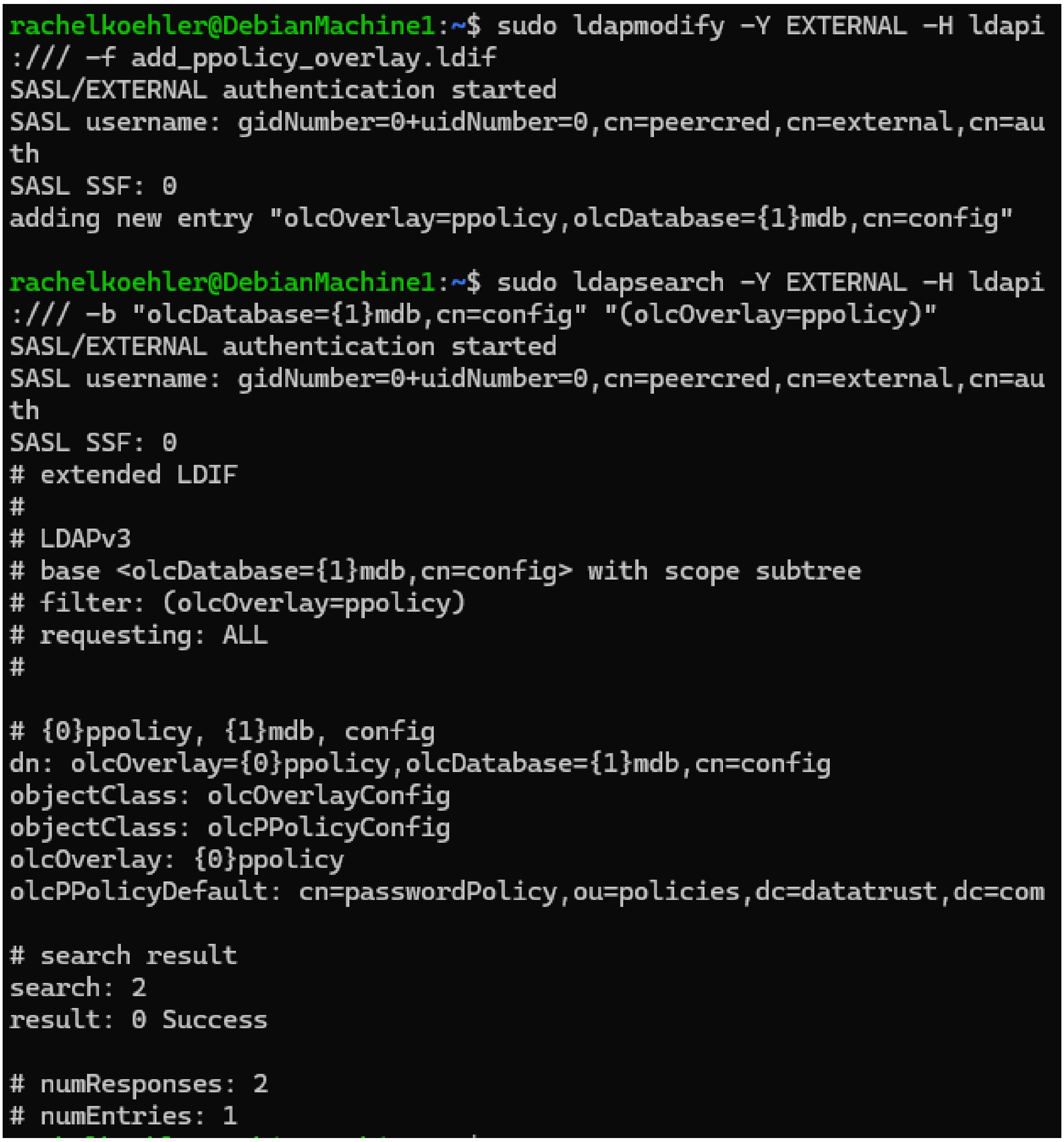
Ici cela ne retourne rien donc besoin de créer et configurer un autre fichier .ldif



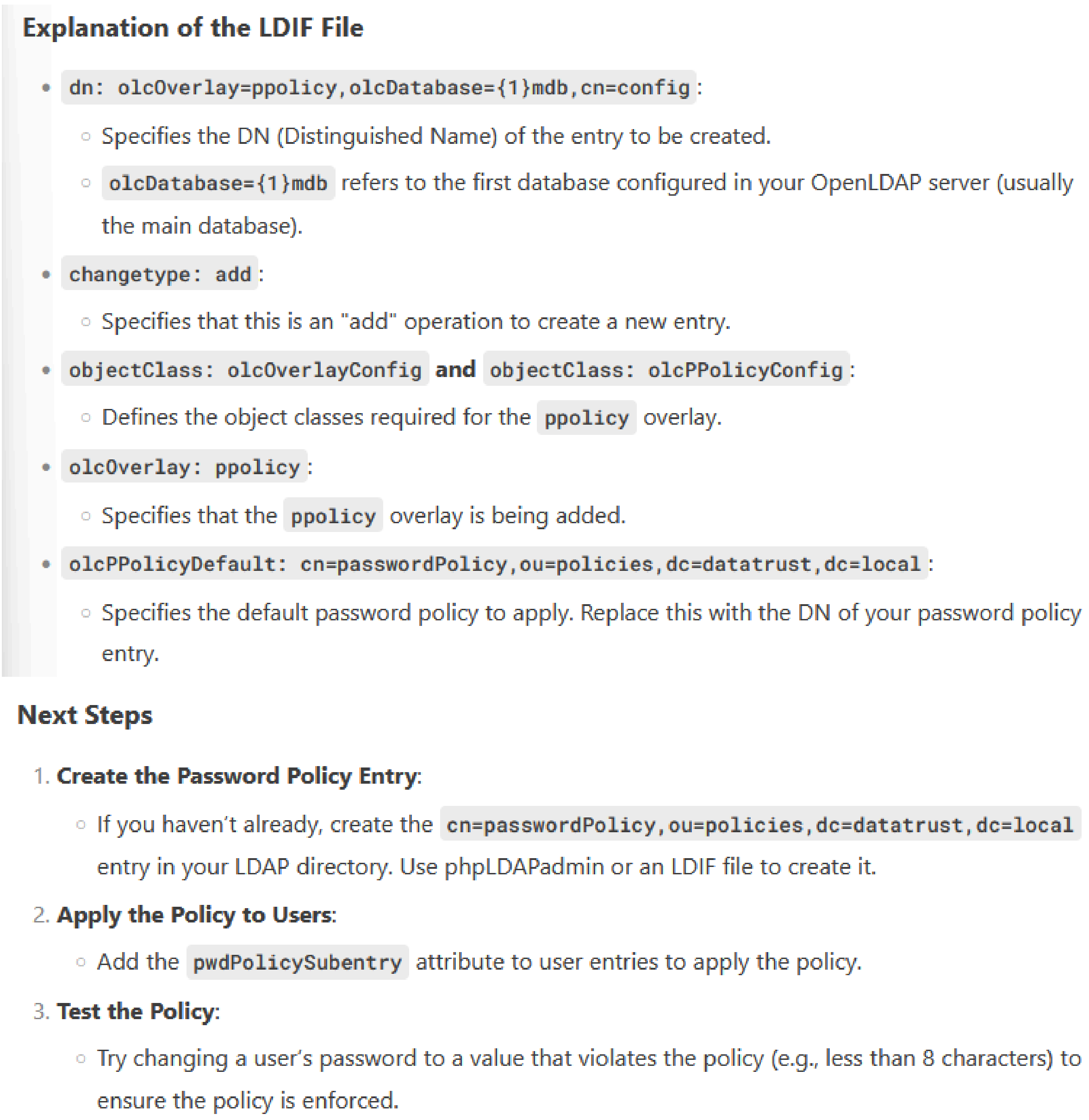
```
rachelkoehler@DebianMachine1:~$ nano add_ppolicy_overlay.ldif
```



```
  GNU nano 7.2            add_ppolicy_overlay.ldif *
dn: olcOverlay=ppolicy,olcDatabase={1}mdb,cn=config
changetype: add
objectClass: olcOverlayConfig
objectClass: olcPPolicyConfig
olcOverlay: ppolicy
olcPPolicyDefault: cn=passwordPolicy,ou=policies,dc=datatrust,dc=com
```



```
rachelkoehler@DebianMachine1:~$ sudo ldapmodify -Y EXTERNAL -H ldapi
:/// -f add_ppolicy_overlay.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=au
th
SASL SSF: 0
adding new entry "olcOverlay=ppolicy,olcDatabase={1}mdb,cn=config"

rachelkoehler@DebianMachine1:~$ sudo ldapsearch -Y EXTERNAL -H ldapi
:/// -b "olcDatabase={1}mdb,cn=config" "(olcOverlay=ppolicy)"
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=au
th
SASL SSF: 0
# extended LDIF
#
# LDAPv3
# base <olcDatabase={1}mdb,cn=config> with scope subtree
# filter: (olcOverlay=ppolicy)
# requesting: ALL
#

# {0}ppolicy, {1}mdb, config
dn: olcOverlay={0}ppolicy,olcDatabase={1}mdb,cn=config
objectClass: olcOverlayConfig
objectClass: olcPPolicyConfig
olcOverlay: {0}ppolicy
olcPPolicyDefault: cn=passwordPolicy,ou=policies,dc=datatrust,dc=com

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

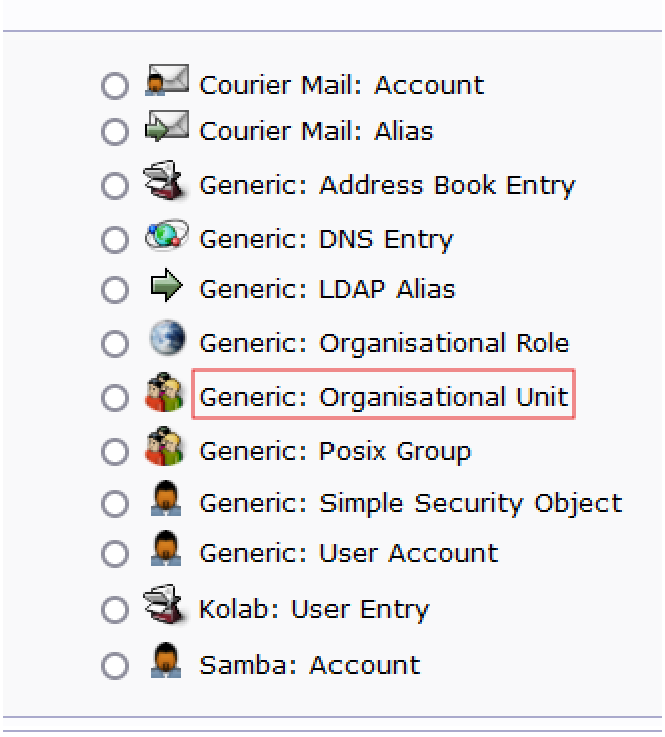


Explanation of the LDIF File

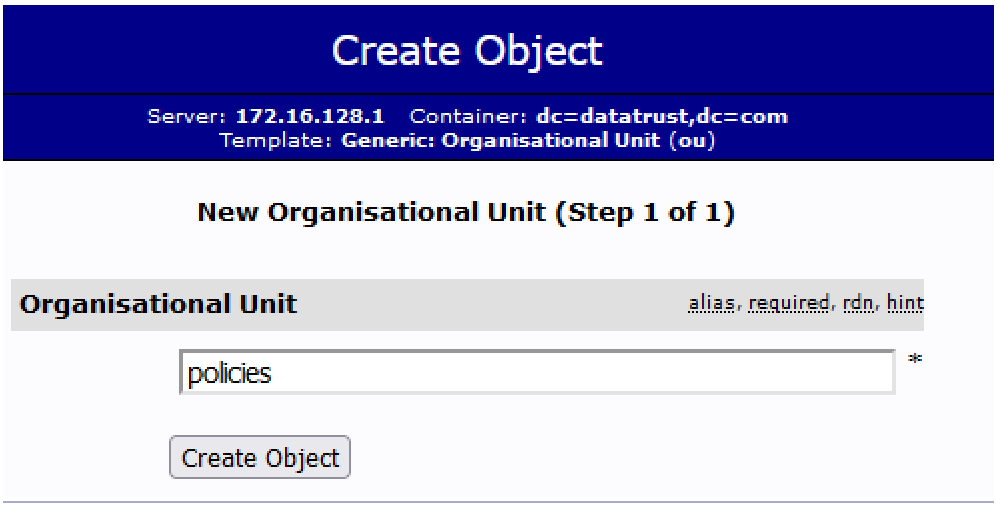
- `dn: olcOverlay=ppolicy,olcDatabase={1}mdb,cn=config`:
  - Specifies the DN (Distinguished Name) of the entry to be created.
  - `olcDatabase={1}mdb` refers to the first database configured in your OpenLDAP server (usually the main database).
- `changetype: add`:
  - Specifies that this is an "add" operation to create a new entry.
- `objectClass: olcOverlayConfig` **and** `objectClass: olcPPolicyConfig`:
  - Defines the object classes required for the `ppolicy` overlay.
- `olcOverlay: ppolicy`:
  - Specifies that the `ppolicy` overlay is being added.
- `olcPPolicyDefault: cn=passwordPolicy,ou=policies,dc=datatrust,dc=local`:
  - Specifies the default password policy to apply. Replace this with the DN of your password policy entry.

Next Steps

1. **Create the Password Policy Entry**:
   - If you haven't already, create the `cn=passwordPolicy,ou=policies,dc=datatrust,dc=local` entry in your LDAP directory. Use phpLDAPadmin or an LDIF file to create it.
2. **Apply the Policy to Users**:
   - Add the `pwdPolicySubentry` attribute to user entries to apply the policy.
3. **Test the Policy**:
   - Try changing a user's password to a value that violates the policy (e.g., less than 8 characters) to ensure the policy is enforced.



○ Courier Mail: Account
○ Courier Mail: Alias
○ Generic: Address Book Entry
○ Generic: DNS Entry
○ Generic: LDAP Alias
○ Generic: Organisational Role
○ [✓ Generic: Organisational Unit]
○ Generic: Posix Group
○ Generic: Simple Security Object
○ Generic: User Account
○ Kolab: User Entry
○ Samba: Account

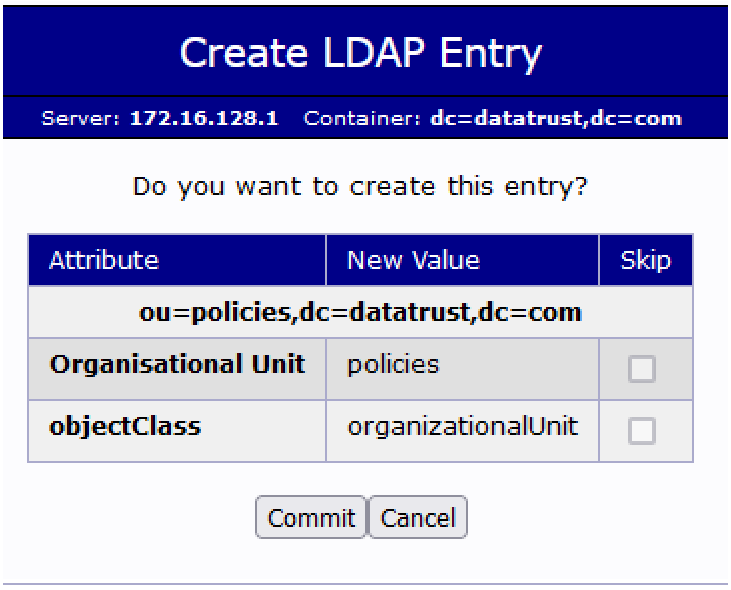


Create Object

Server: **172.16.128.1** Container: **dc=datatrust,dc=com**
Template: **Generic: Organisational Unit** (ou)

New Organisational Unit (Step 1 of 1)

Organisational Unit	alias, required, rdn, hint
	policies *

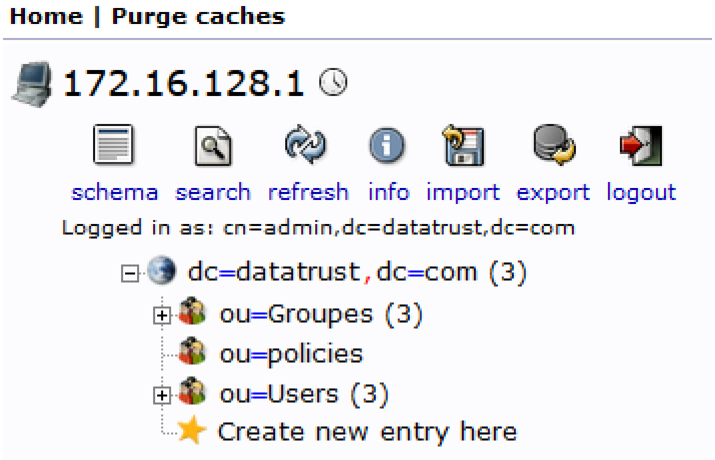


Create LDAP Entry

Server: **172.16.128.1** Container: **dc=datatrust,dc=com**

Do you want to create this entry?

Attribute	New Value	Skip
ou=policies,dc=datatrust,dc=com		
Organisational Unit	policies	□
objectClass	organizationalUnit	□



Logged in as: cn=admin,dc=datatrust,dc=com

```
dc=datatrust,dc=com (3)
    ou=Groupes (3)
    ou=policies
    ou=Users (3)
    Create new entry here
```

Then we ends our collaboration with phpLDAPAdmin :

🔥 Conclusion

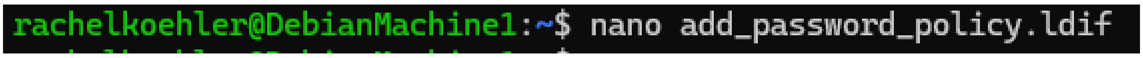
✔ **phpLDAPadmin isn't ideal for this task**, so we use **LDIF files** instead.

✔ **Manually create a password policy** and **import it into LDAP**.

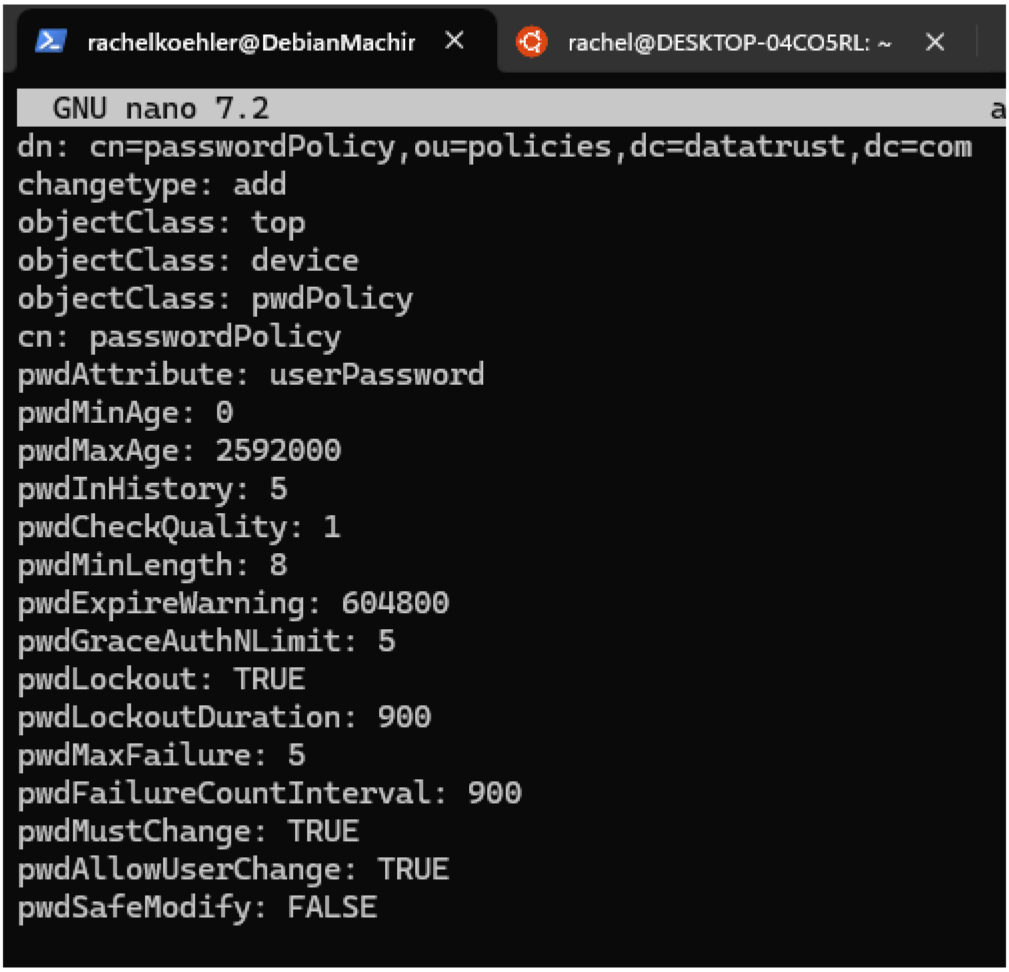
✔ **Link the policy to OpenLDAP** using `olcPPolicyDefault`.

✔ **Verify using** `ldapsearch`.

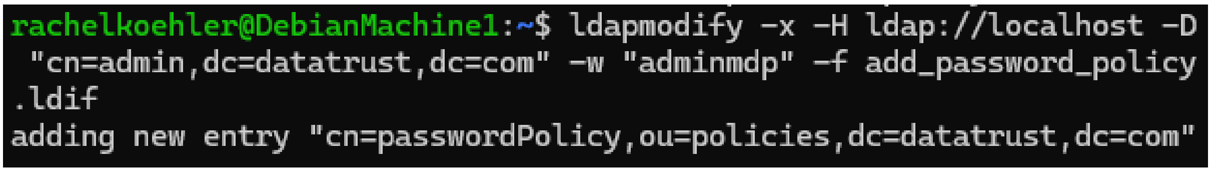
So now it's full LDIF file management:



```
rachelkoehler@DebianMachine1:~$ nano add_password_policy.ldif
```

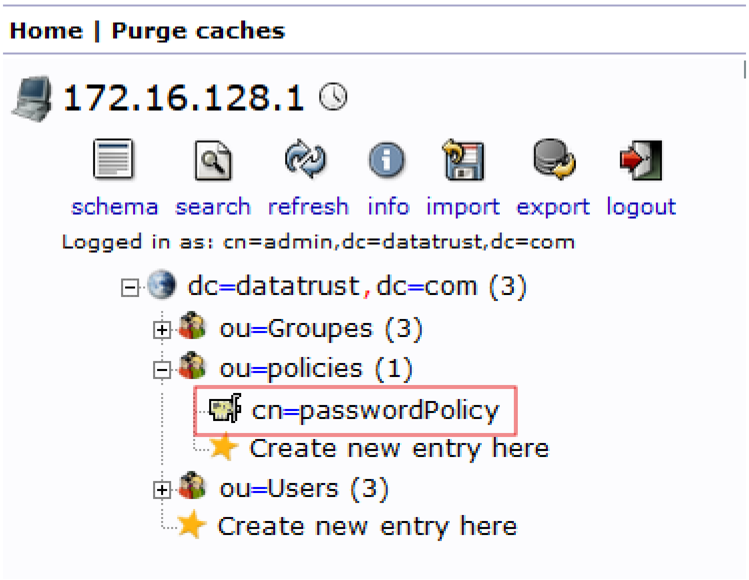


```
  GNU nano 7.2                                            a
dn: cn=passwordPolicy,ou=policies,dc=datatrust,dc=com
changetype: add
objectClass: top
objectClass: device
objectClass: pwdPolicy
cn: passwordPolicy
pwdAttribute: userPassword
pwdMinAge: 0
pwdMaxAge: 2592000
pwdInHistory: 5
pwdCheckQuality: 1
pwdMinLength: 8
pwdExpireWarning: 604800
pwdGraceAuthNLimit: 5
pwdLockout: TRUE
pwdLockoutDuration: 900
pwdMaxFailure: 5
pwdFailureCountInterval: 900
pwdMustChange: TRUE
pwdAllowUserChange: TRUE
pwdSafeModify: FALSE
```



```
rachelkoehler@DebianMachine1:~$ ldapmodify -x -H ldap://localhost -D
 "cn=admin,dc=datatrust,dc=com" -w "adminmdp" -f add_password_policy
.ldif
adding new entry "cn=passwordPolicy,ou=policies,dc=datatrust,dc=com"
```

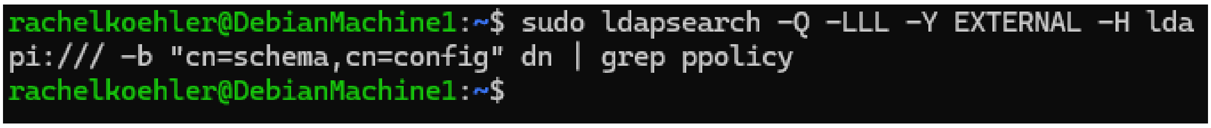
TADAM !



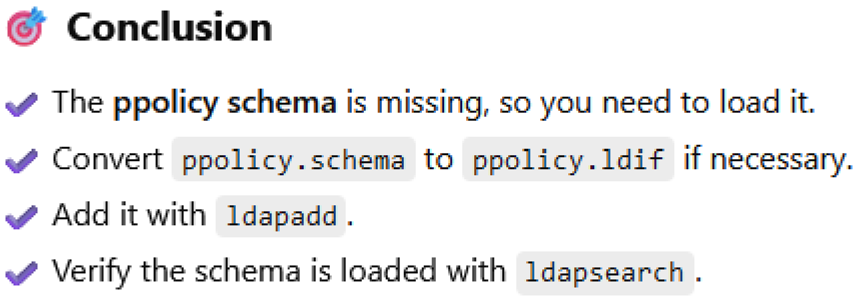
Logged in as: cn=admin,dc=datatrust,dc=com

```
⊟ dc=datatrust,dc=com (3)
  ⊞ ou=Groupes (3)
  ⊟ ou=policies (1)
      cn=passwordPolicy
      Create new entry here
  ⊞ ou=Users (3)
  Create new entry here
```

Before delving deeper let's checl if ppolicy is well loaded:



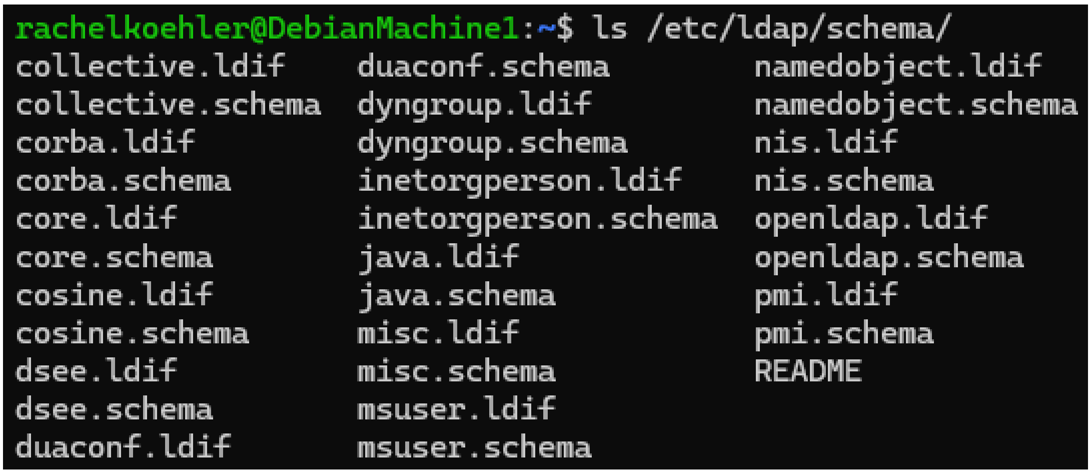
```
rachelkoehler@DebianMachine1:~$ sudo ldapsearch -Q -LLL -Y EXTERNAL -H lda
pi:/// -b "cn=schema,cn=config" dn | grep ppolicy
rachelkoehler@DebianMachine1:~$
```



Conclusion

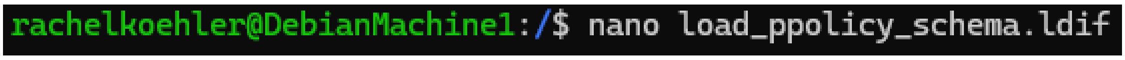
- ✓ The **ppolicy schema** is missing, so you need to load it.
- ✓ Convert `ppolicy.schema` to `ppolicy.ldif` if necessary.
- ✓ Add it with `ldapadd`.
- ✓ Verify the schema is loaded with `ldapsearch`.

Vérifier la présence du fichier ppolicy.ldif ds chemin d'accès particulier

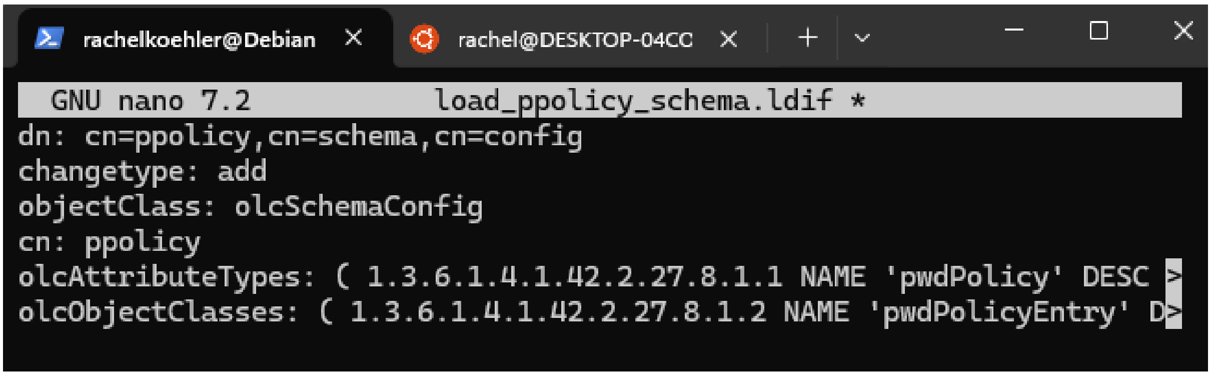


```
rachelkoehler@DebianMachine1:~$ ls /etc/ldap/schema/
collective.ldif    duaconf.schema       namedobject.ldif
collective.schema  dyngroup.ldif        namedobject.schema
corba.ldif         dyngroup.schema      nis.ldif
corba.schema       inetorgperson.ldif   nis.schema
core.ldif          inetorgperson.schema openldap.ldif
core.schema        java.ldif            openldap.schema
cosine.ldif        java.schema          pmi.ldif
cosine.schema      misc.ldif            pmi.schema
dsee.ldif          misc.schema          README
dsee.schema        msuser.ldif
duaconf.ldif       msuser.schema
```

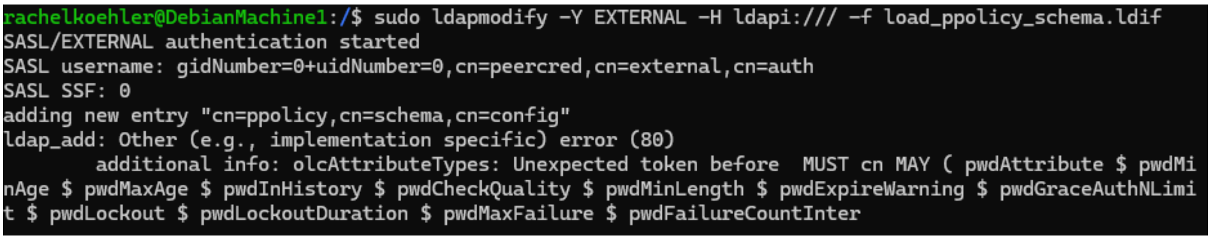
Create manually the schema:



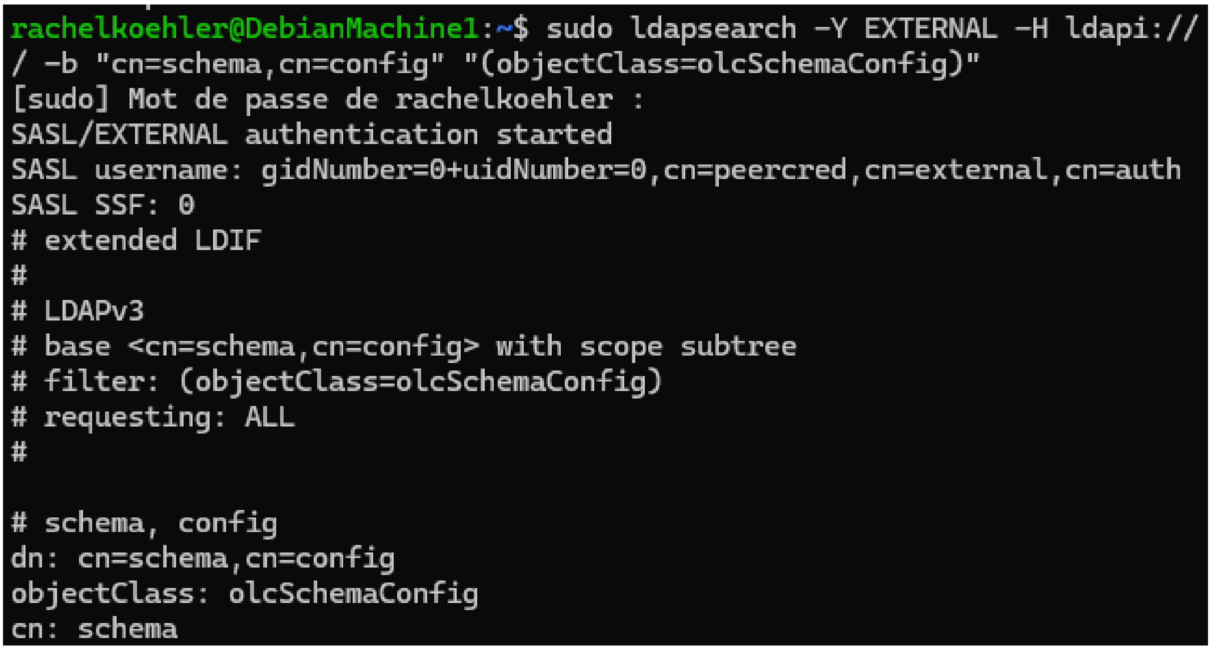
```
rachelkoehler@DebianMachine1:/$ nano load_ppolicy_schema.ldif
```



```
  GNU nano 7.2              load_ppolicy_schema.ldif *
dn: cn=ppolicy,cn=schema,cn=config
changetype: add
objectClass: olcSchemaConfig
cn: ppolicy
olcAttributeTypes: ( 1.3.6.1.4.1.42.2.27.8.1.1 NAME 'pwdPolicy' DESC >
olcObjectClasses: ( 1.3.6.1.4.1.42.2.27.8.1.2 NAME 'pwdPolicyEntry' D>
```

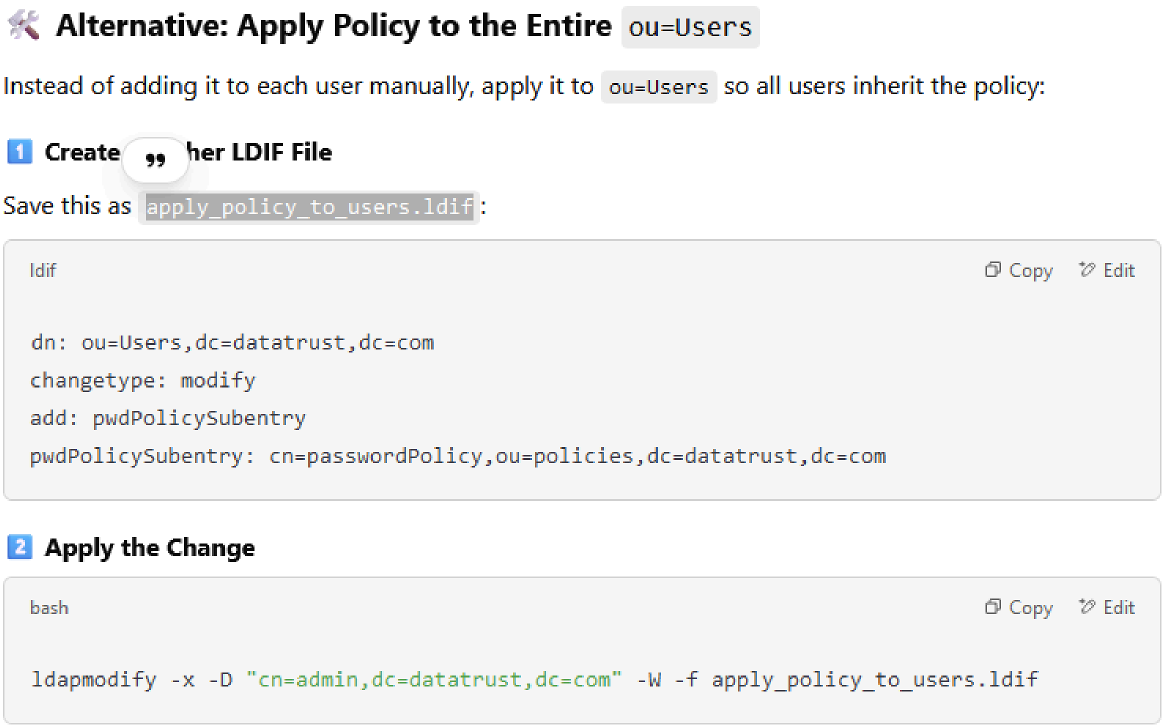


```
rachelkoehler@DebianMachine1:/$ sudo ldapmodify -Y EXTERNAL -H ldapi:/// -f load_ppolicy_schema.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=ppolicy,cn=schema,cn=config"
ldap_add: Other (e.g., implementation specific) error (80)
        additional info: olcAttributeTypes: Unexpected token before  MUST cn MAY ( pwdAttribute $ pwdMi
nAge $ pwdMaxAge $ pwdInHistory $ pwdCheckQuality $ pwdMinLength $ pwdExpireWarning $ pwdGraceAuthNLimi
t $ pwdLockout $ pwdLockoutDuration $ pwdMaxFailure $ pwdFailureCountInter
```



```
rachelkoehler@DebianMachine1:~$ sudo ldapsearch -Y EXTERNAL -H ldapi://
/ -b "cn=schema,cn=config" "(objectClass=olcSchemaConfig)"
[sudo] Mot de passe de rachelkoehler :
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
# extended LDIF
#
# LDAPv3
# base <cn=schema,cn=config> with scope subtree
# filter: (objectClass=olcSchemaConfig)
# requesting: ALL
#

# schema, config
dn: cn=schema,cn=config
objectClass: olcSchemaConfig
cn: schema
```

Sur phpadmin (pas vraiment)



🛠️ Alternative: Apply Policy to the Entire `ou=Users`

Instead of adding it to each user manually, apply it to `ou=Users` so all users inherit the policy:

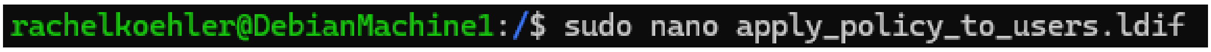
1️⃣ Create ❞ her LDIF File

Save this as `apply_policy_to_users.ldif`:

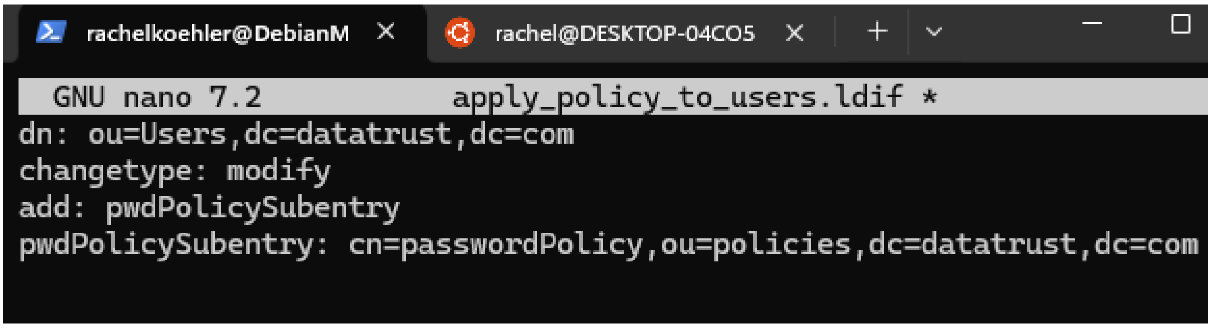
```ldif
dn: ou=Users,dc=datatrust,dc=com
changetype: modify
add: pwdPolicySubentry
pwdPolicySubentry: cn=passwordPolicy,ou=policies,dc=datatrust,dc=com
```

2️⃣ Apply the Change

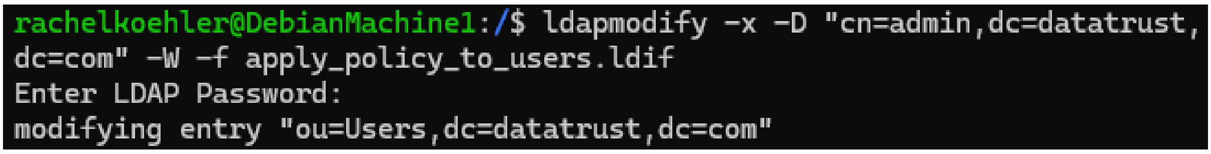
```bash
ldapmodify -x -D "cn=admin,dc=datatrust,dc=com" -W -f apply_policy_to_users.ldif
```



```
rachelkoehler@DebianMachine1:/$ sudo nano apply_policy_to_users.ldif
```

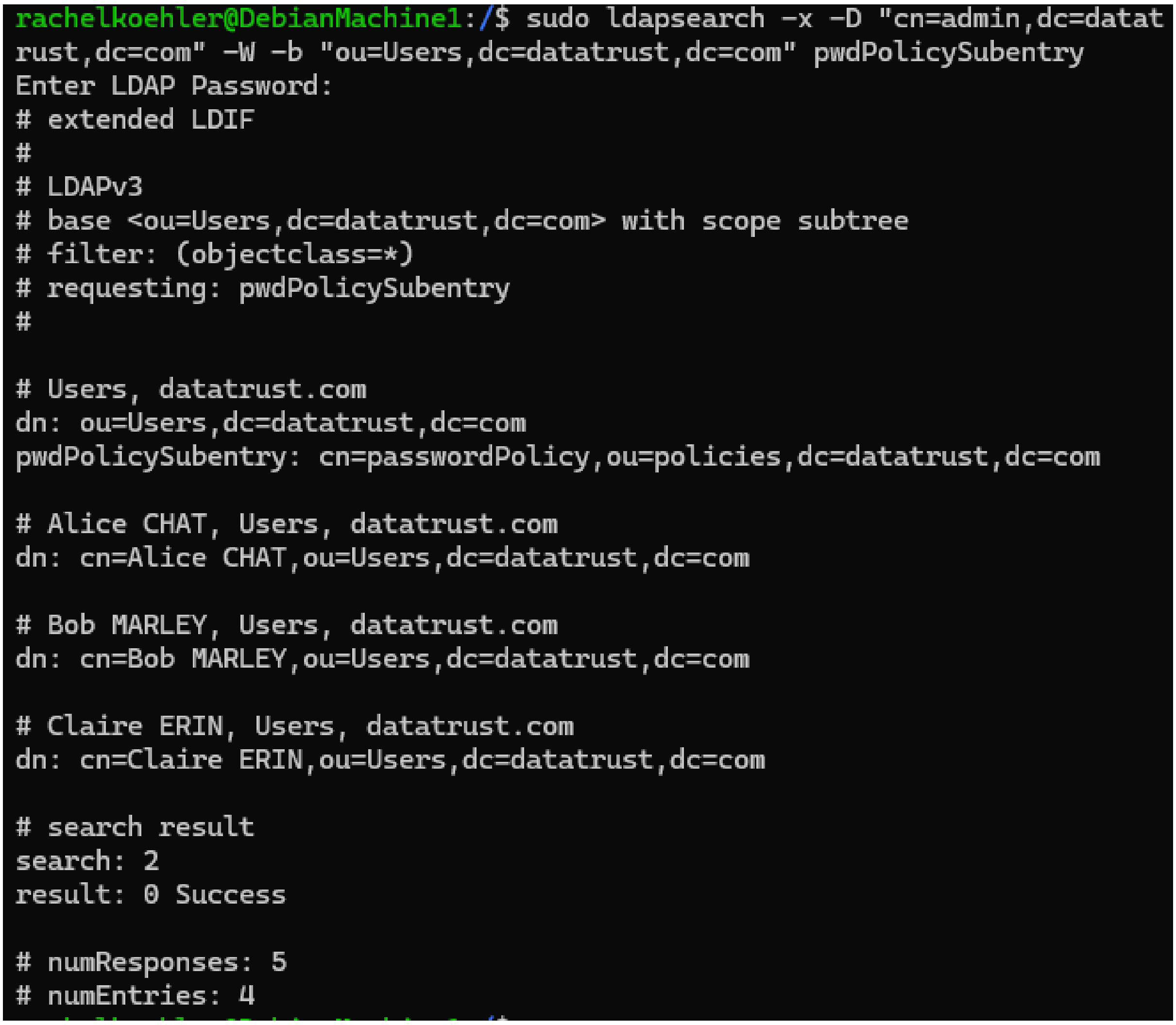


```
  GNU nano 7.2               apply_policy_to_users.ldif *
dn: ou=Users,dc=datatrust,dc=com
changetype: modify
add: pwdPolicySubentry
pwdPolicySubentry: cn=passwordPolicy,ou=policies,dc=datatrust,dc=com
```



```
rachelkoehler@DebianMachine1:/$ ldapmodify -x -D "cn=admin,dc=datatrust,
dc=com" -W -f apply_policy_to_users.ldif
Enter LDAP Password:
modifying entry "ou=Users,dc=datatrust,dc=com"
```

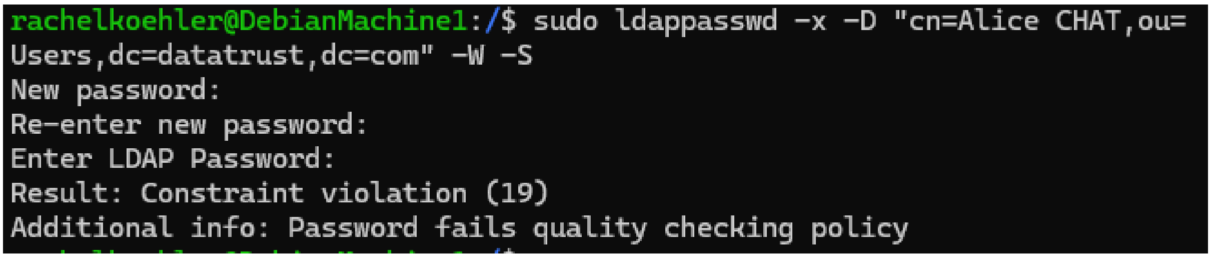
Vérifier que la politique est appliquée



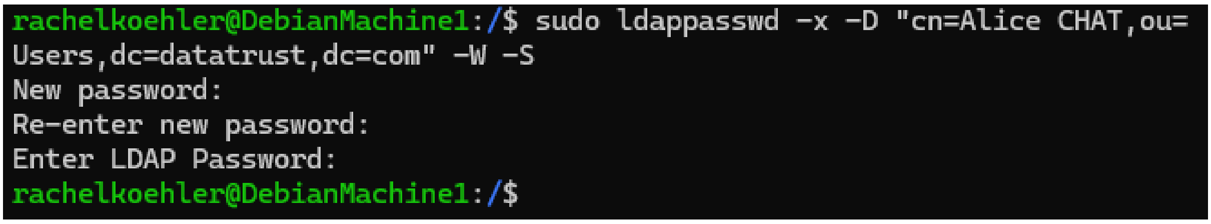
```
rachelkoehler@DebianMachine1:/$ sudo ldapsearch -x -D "cn=admin,dc=datat
rust,dc=com" -W -b "ou=Users,dc=datatrust,dc=com" pwdPolicySubentry
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <ou=Users,dc=datatrust,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: pwdPolicySubentry
#

# Users, datatrust.com
dn: ou=Users,dc=datatrust,dc=com
pwdPolicySubentry: cn=passwordPolicy,ou=policies,dc=datatrust,dc=com

# Alice CHAT, Users, datatrust.com
dn: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com

# Bob MARLEY, Users, datatrust.com
dn: cn=Bob MARLEY,ou=Users,dc=datatrust,dc=com

# Claire ERIN, Users, datatrust.com
dn: cn=Claire ERIN,ou=Users,dc=datatrust,dc=com

# search result
search: 2
result: 0 Success

# numResponses: 5
# numEntries: 4
```

Your output confirms that `pwdPolicySubentry` **is now set at the** `ou=Users` **level**, but **it is not appearing under individual users** (`Alice CHAT` , `Bob MARLEY` , `Claire ERIN`).

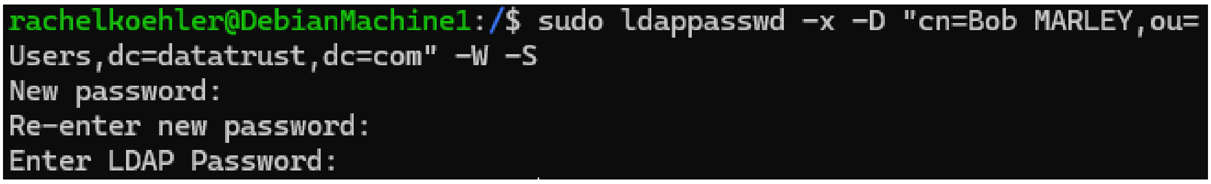
This means that the password policy is applied at the **OU level**, but OpenLDAP **does not automatically display inherited attributes** at the user level. However, the policy **should still be working.**



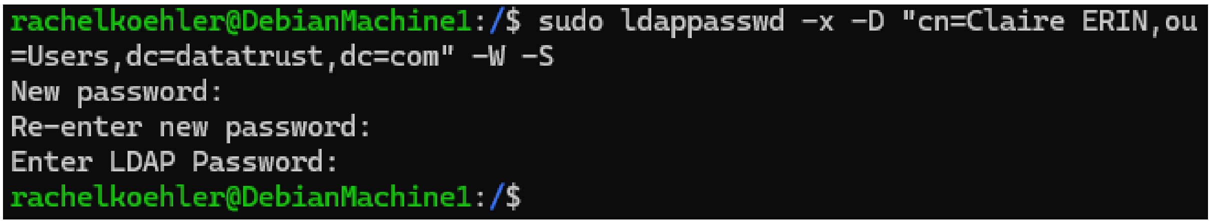
```
rachelkoehler@DebianMachine1:/$ sudo ldappasswd -x -D "cn=Alice CHAT,ou=
Users,dc=datatrust,dc=com" -W -S
New password:
Re-enter new password:
Enter LDAP Password:
Result: Constraint violation (19)
Additional info: Password fails quality checking policy
```



```
rachelkoehler@DebianMachine1:/$ sudo ldappasswd -x -D "cn=Alice CHAT,ou=
Users,dc=datatrust,dc=com" -W -S
New password:
Re-enter new password:
Enter LDAP Password:
rachelkoehler@DebianMachine1:/$
```



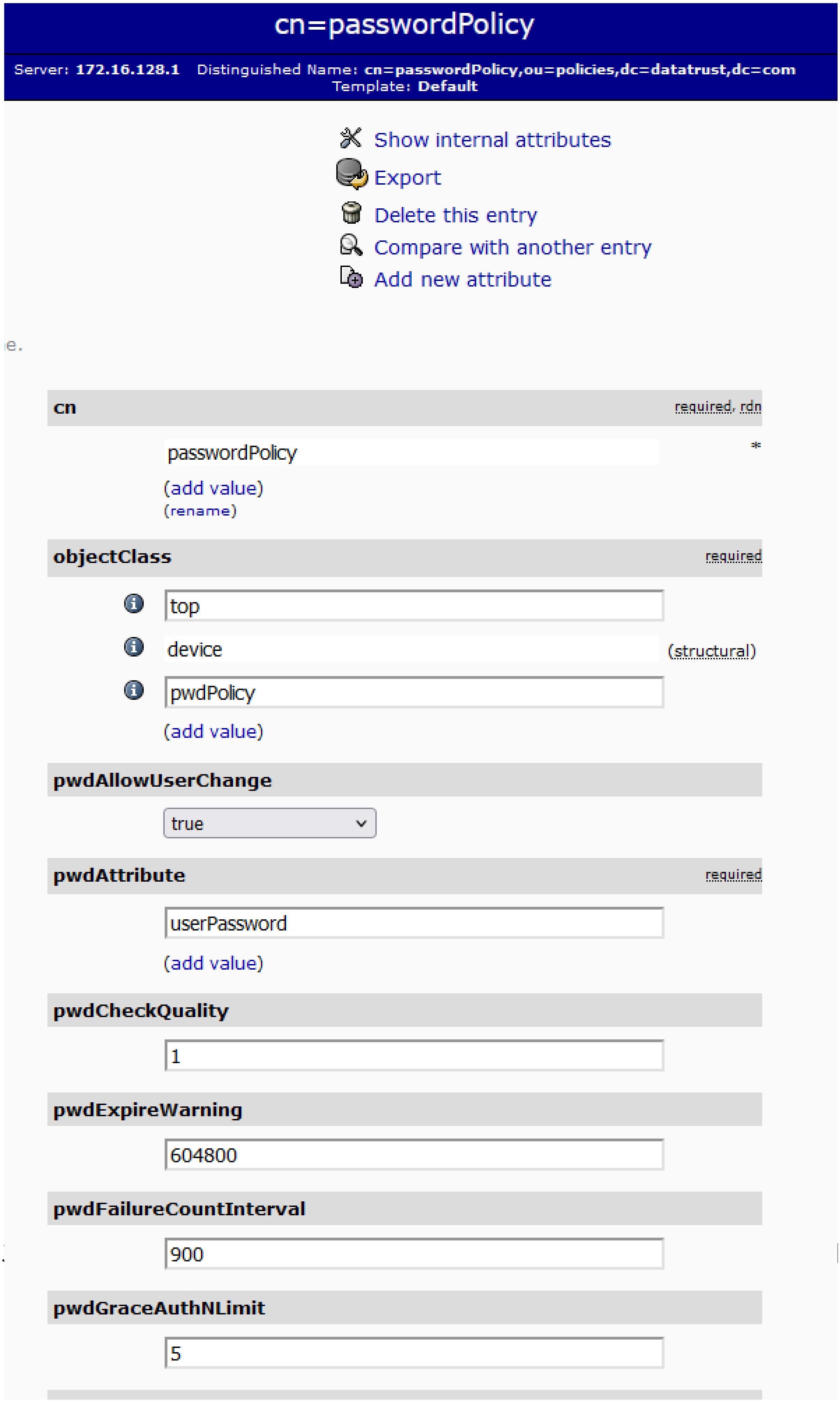
```
rachelkoehler@DebianMachine1:/$ sudo ldappasswd -x -D "cn=Bob MARLEY,ou=
Users,dc=datatrust,dc=com" -W -S
New password:
Re-enter new password:
Enter LDAP Password:
```



```
rachelkoehler@DebianMachine1:/$ sudo ldappasswd -x -D "cn=Claire ERIN,ou
=Users,dc=datatrust,dc=com" -W -S
New password:
Re-enter new password:
Enter LDAP Password:
rachelkoehler@DebianMachine1:/$
```

Changes that you can see in the phadmininterface:

L'apparition des attributs



cn=passwordPolicy

Server: **172.16.128.1** Distinguished Name: **cn=passwordPolicy,ou=policies,dc=datatrust,dc=com**
Template: **Default**

e.

cn required, rdn

passwordPolicy *

objectClass required

top

device (structural)

pwdPolicy

pwdAllowUserChange

true

pwdAttribute required

userPassword

pwdCheckQuality

1

pwdExpireWarning

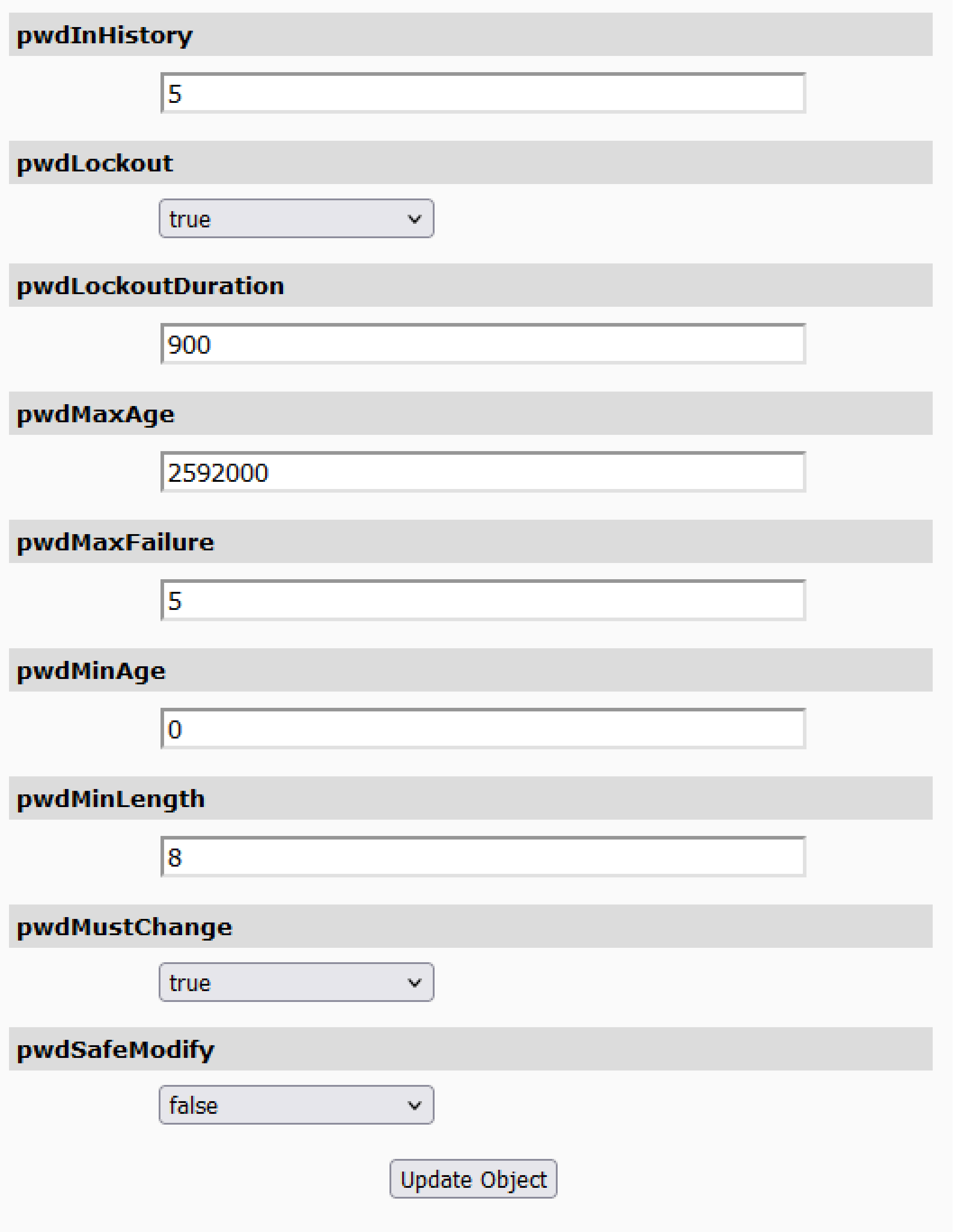
604800

pwdFailureCountInterval

900

pwdGraceAuthNLimit

5



pwdInHistory: 5

pwdLockout: true

pwdLockoutDuration: 900

pwdMaxAge: 2592000

pwdMaxFailure: 5

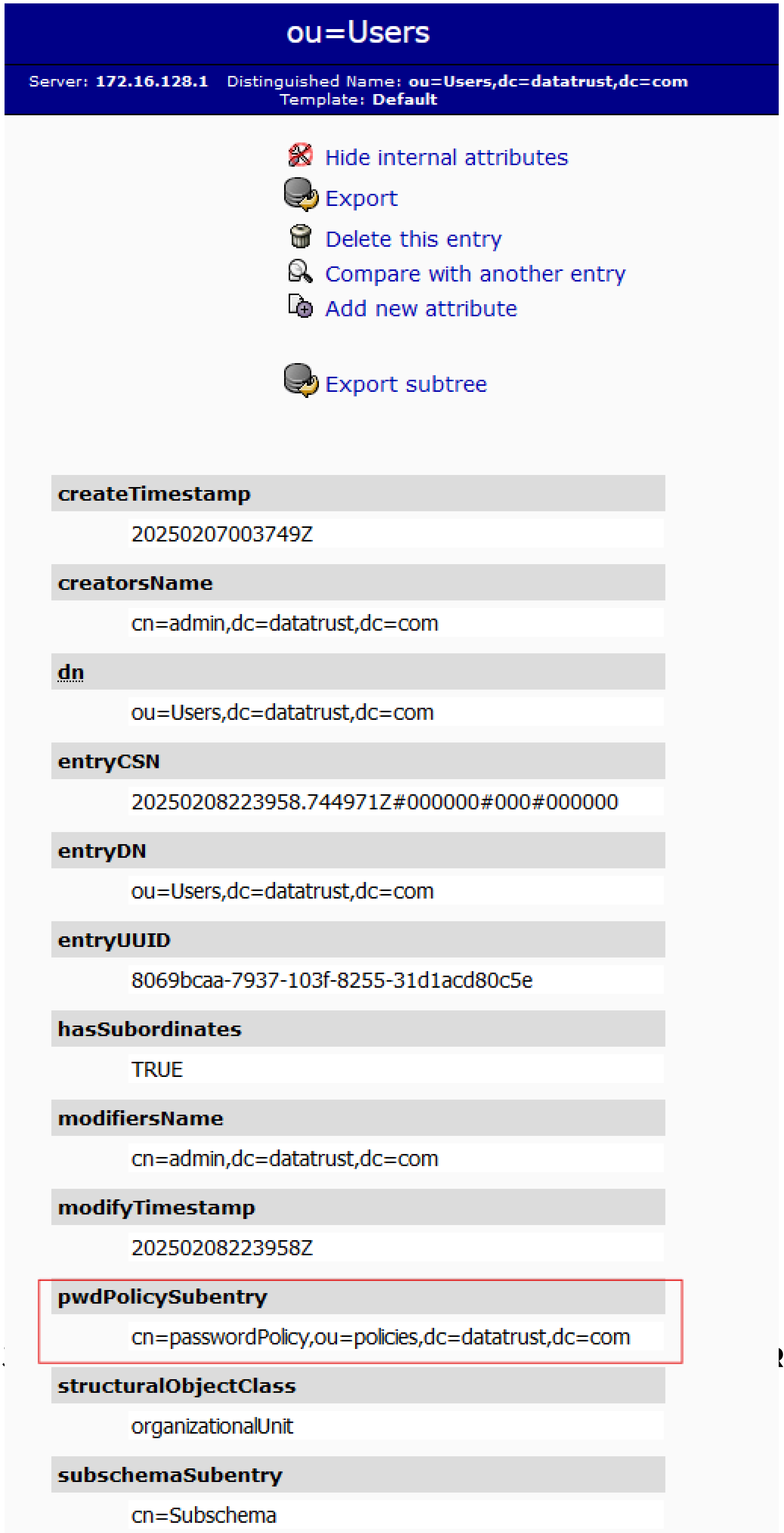
pwdMinAge: 0

pwdMinLength: 8

pwdMustChange: true

pwdSafeModify: false

Chnages at ou=USERs level:



ou=Users

Server: **172.16.128.1** Distinguished Name: **ou=Users,dc=datatrust,dc=com**
Template: **Default**

createTimestamp
20250207003749Z

creatorsName
cn=admin,dc=datatrust,dc=com

dn
ou=Users,dc=datatrust,dc=com

entryCSN
20250208223958.744971Z#000000#000#000000

entryDN
ou=Users,dc=datatrust,dc=com

entryUUID
8069bcaa-7937-103f-8255-31d1acd80c5e

hasSubordinates
TRUE

modifiersName
cn=admin,dc=datatrust,dc=com

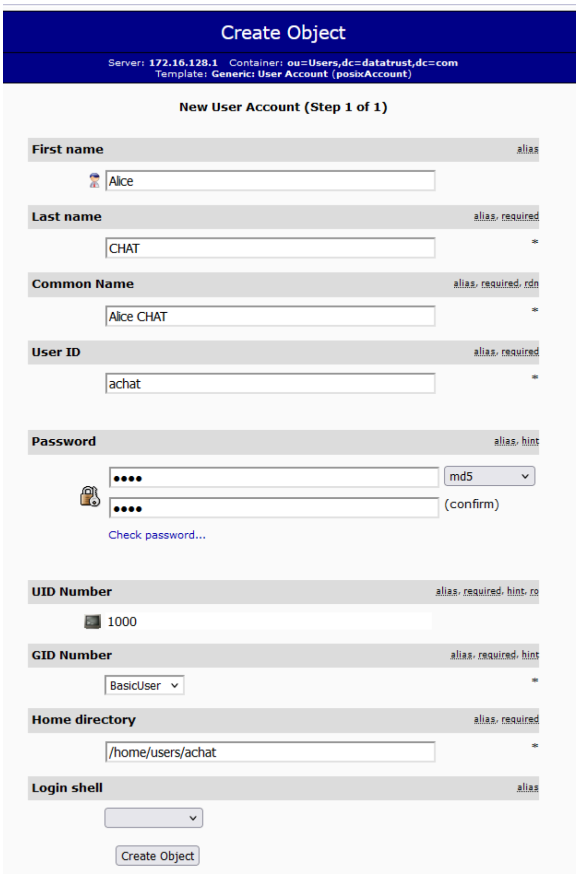
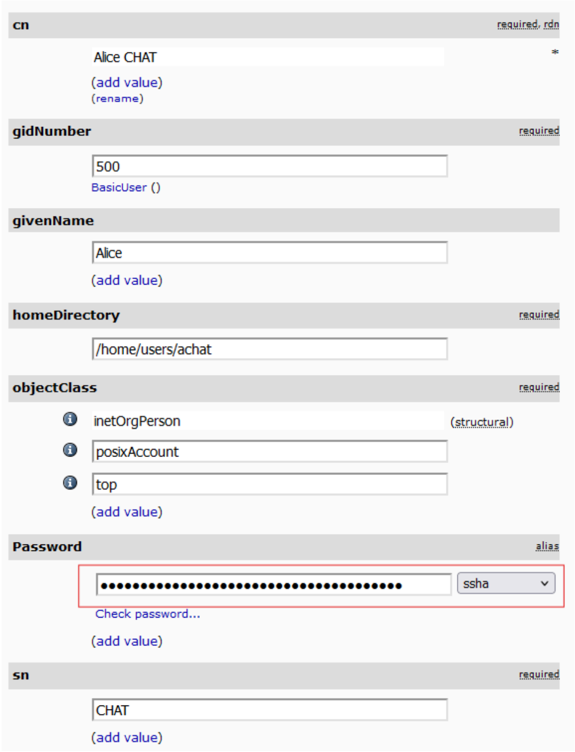
modifyTimestamp
20250208223958Z

pwdPolicySubentry
cn=passwordPolicy,ou=policies,dc=datatrust,dc=com

structuralObjectClass
organizationalUnit

subschemaSubentry
cn=Subschema

Mise à jour du mot de passe :

AVANT (format md5 et 4 malheureux petits points)	**APRÈS** (+ pints et ssha)
	

Create Object

Server: 172.16.128.1 Container: ou=Users,dc=datatrust,dc=com
Template: Generic: User Account (posixAccount)

New User Account (Step 1 of 1)

First name alias
Alice

Last name alias, required
CHAT *

Common Name alias, required, rdn
Alice CHAT *

User ID alias, required
achat *

Password alias, hint
•••• md5
•••• (confirm)
Check password...

UID Number alias, required, hint, ro
1000

GID Number alias, required, hint
BasicUser *

Home directory alias, required
/home/users/achat *

Login shell alias

Create Object

cn required, rdn
Alice CHAT *
(add value)
(rename)

gidNumber required
500
BasicUser ()

givenName
Alice
(add value)

homeDirectory required
/home/users/achat

objectClass required
inetOrgPerson (structural)
posixAccount
top
(add value)

Password alias
••••••••••••••••••••••••••••••••••••••• ssha
Check password...
(add value)

sn required
CHAT
(add value)

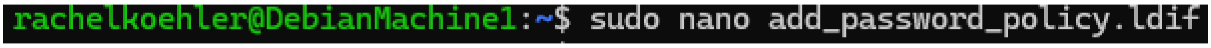
2. Déconnexion Automatique

Recherchez comment mettre en place une déconnexion automatique après une période d'inactivité pour sécuriser les sessions utilisateur.

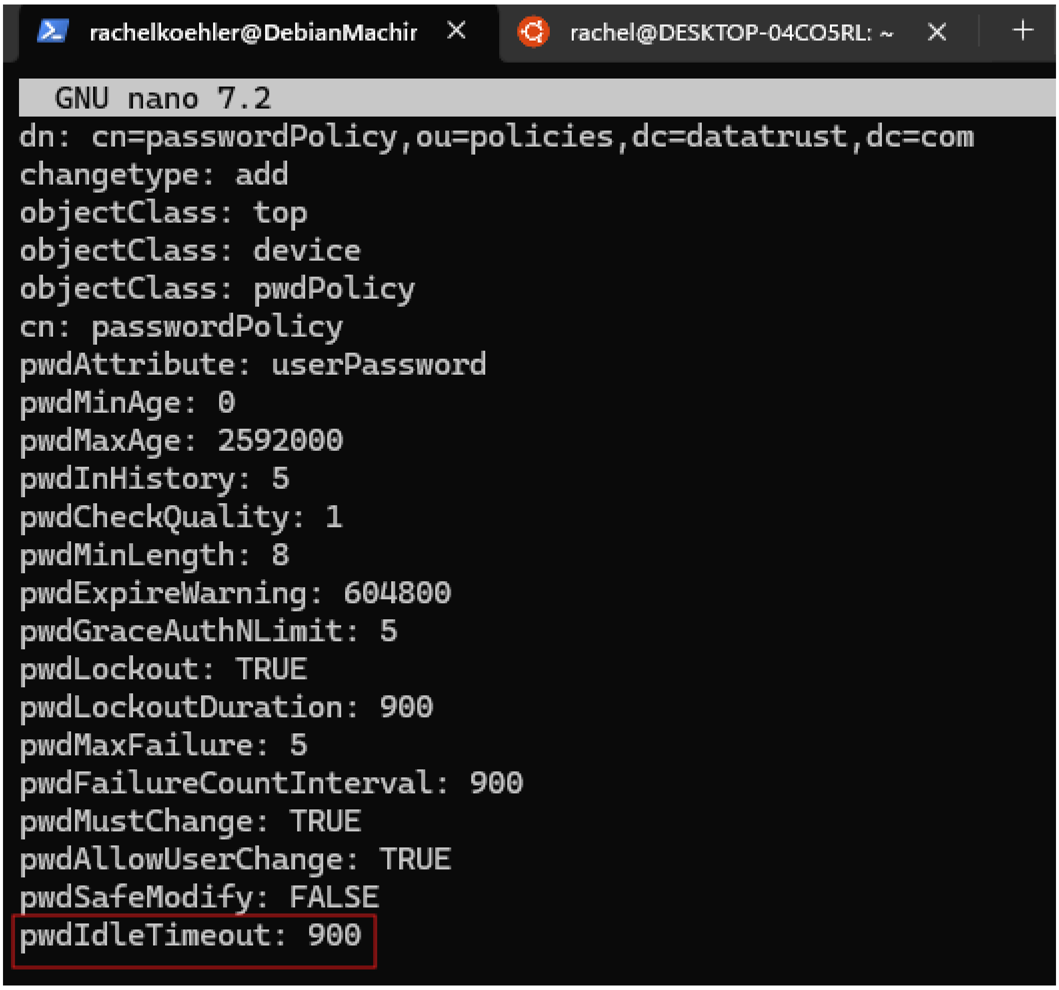
Configurer un "Idle Timeout" :

- *Ajouter une politique de déconnexion automatique en utilisant l'attribut* `pwdIdleTimeout` *ds e fichier de config mdp :* ***add_password_policy.ldif***

`pwdIdleTimeout` *: Temps d'inactivité en secondes (ici, 900 secondes = 15 minutes).*

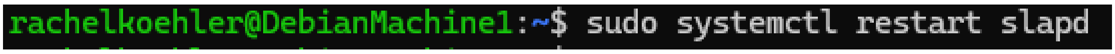


```
rachelkoehler@DebianMachine1:~$ sudo nano add_password_policy.ldif
```



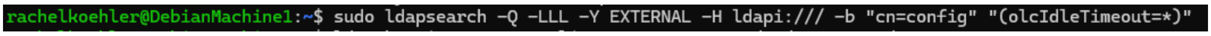
```
  GNU nano 7.2
dn: cn=passwordPolicy,ou=policies,dc=datatrust,dc=com
changetype: add
objectClass: top
objectClass: device
objectClass: pwdPolicy
cn: passwordPolicy
pwdAttribute: userPassword
pwdMinAge: 0
pwdMaxAge: 2592000
pwdInHistory: 5
pwdCheckQuality: 1
pwdMinLength: 8
pwdExpireWarning: 604800
pwdGraceAuthNLimit: 5
pwdLockout: TRUE
pwdLockoutDuration: 900
pwdMaxFailure: 5
pwdFailureCountInterval: 900
pwdMustChange: TRUE
pwdAllowUserChange: TRUE
pwdSafeModify: FALSE
pwdIdleTimeout: 900
```

On redémarre le système



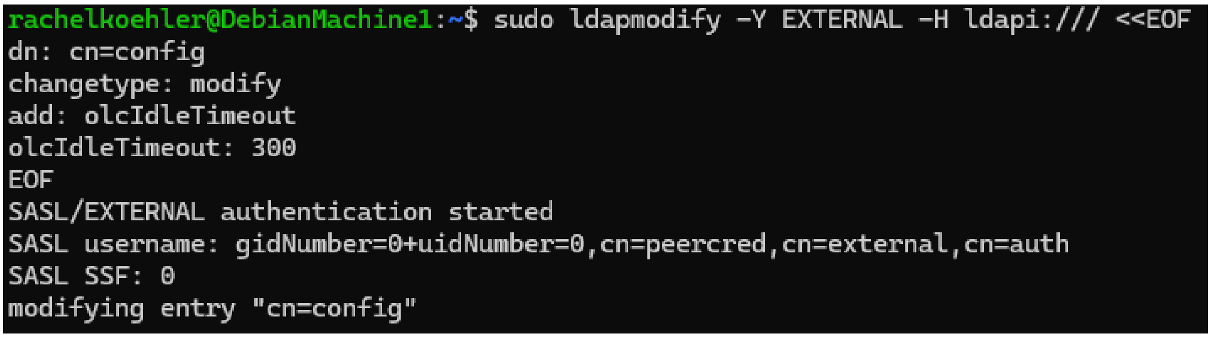
```
rachelkoehler@DebianMachine1:~$ sudo systemctl restart slapd
```

Vérifier que “olcIdleTimeout” soit bien configuré:



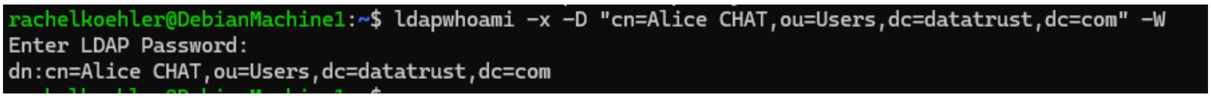
```
rachelkoehler@DebianMachine1:~$ sudo ldapsearch -Q -LLL -Y EXTERNAL -H ldapi:/// -b "cn=config" "(olcIdleTimeout=*)"
```

Si rien n'apparait doit configurer “olcIdleTimeout”

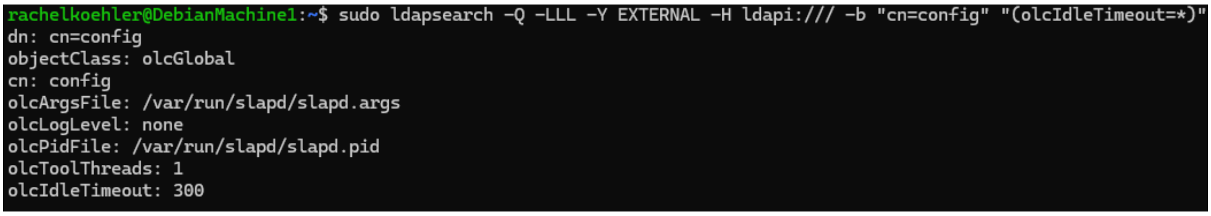


```
rachelkoehler@DebianMachine1:~$ sudo ldapmodify -Y EXTERNAL -H ldapi:/// <<EOF
dn: cn=config
changetype: modify
add: olcIdleTimeout
olcIdleTimeout: 300
EOF
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
modifying entry "cn=config"
```

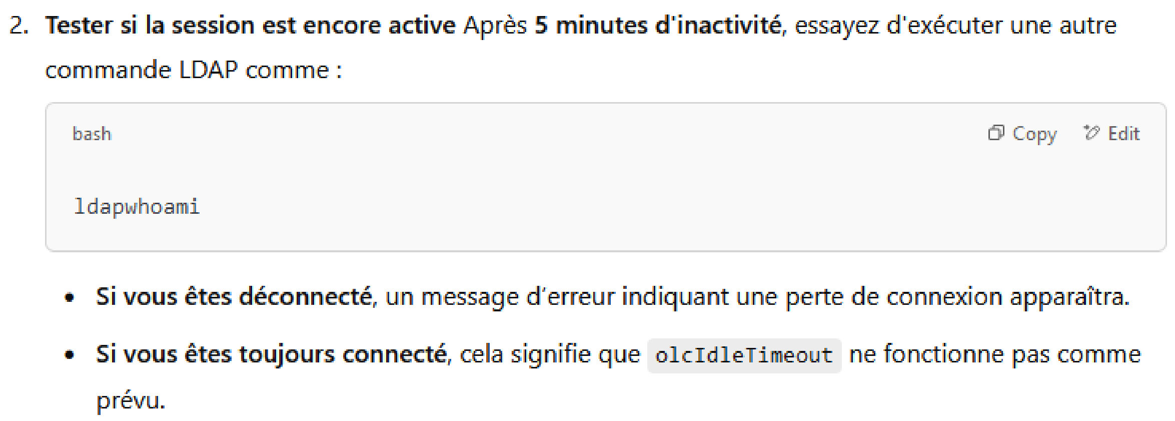
Tester la connexion automatique sur CLI pour Alice:



```
rachelkoehler@DebianMachine1:~$ ldapwhoami -x -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com" -W
Enter LDAP Password:
dn:cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
```



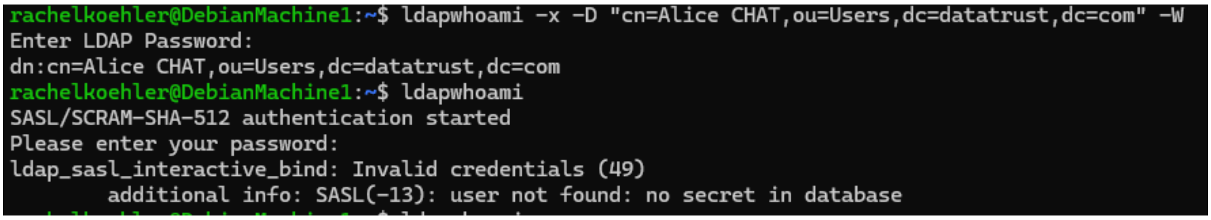
```
rachelkoehler@DebianMachine1:~$ sudo ldapsearch -Q -LLL -Y EXTERNAL -H ldapi:/// -b "cn=config" "(olcIdleTimeout=*)"
dn: cn=config
objectClass: olcGlobal
cn: config
olcArgsFile: /var/run/slapd/slapd.args
olcLogLevel: none
olcPidFile: /var/run/slapd/slapd.pid
olcToolThreads: 1
olcIdleTimeout: 300
```



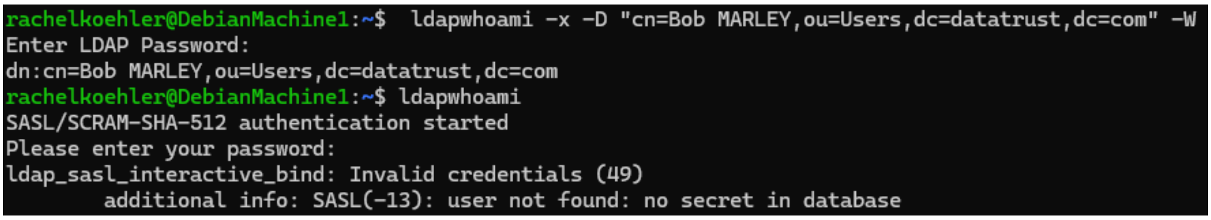
2. **Tester si la session est encore active** Après **5 minutes d'inactivité**, essayez d'exécuter une autre commande LDAP comme :

```bash
ldapwhoami
```

- **Si vous êtes déconnecté**, un message d'erreur indiquant une perte de connexion apparaîtra.
- **Si vous êtes toujours connecté**, cela signifie que `olcIdleTimeout` ne fonctionne pas comme prévu.



```
rachelkoehler@DebianMachine1:~$ ldapwhoami -x -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com" -W
Enter LDAP Password:
dn:cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
rachelkoehler@DebianMachine1:~$ ldapwhoami
SASL/SCRAM-SHA-512 authentication started
Please enter your password:
ldap_sasl_interactive_bind: Invalid credentials (49)
        additional info: SASL(-13): user not found: no secret in database
```



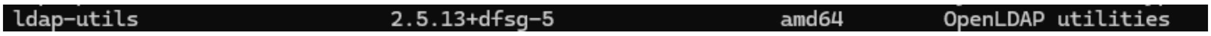
```
rachelkoehler@DebianMachine1:~$  ldapwhoami -x -D "cn=Bob MARLEY,ou=Users,dc=datatrust,dc=com" -W
Enter LDAP Password:
dn:cn=Bob MARLEY,ou=Users,dc=datatrust,dc=com
rachelkoehler@DebianMachine1:~$ ldapwhoami
SASL/SCRAM-SHA-512 authentication started
Please enter your password:
ldap_sasl_interactive_bind: Invalid credentials (49)
        additional info: SASL(-13): user not found: no secret in database
```

Tester la connexion automatique sur phpLDAPadmin pour Claire:

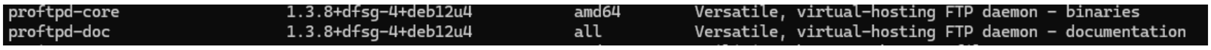
BONUS ! Pour aller plus loin

Pour approfondir, vous avez la possibilité d'utiliser le projet précédent « DNS – DHCP – FTP » que vous avez réalisé.
Sur le serveur FTP, utilisez LDAP pour les identifiants de connexion et les permissions d'accès aux dossiers spécifiques.

Vérifier l’installation des paquet requis:

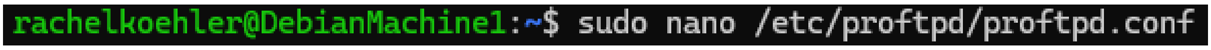


```
ldap-utils                    2.5.13+dfsg-5                    amd64        OpenLDAP utilities
```

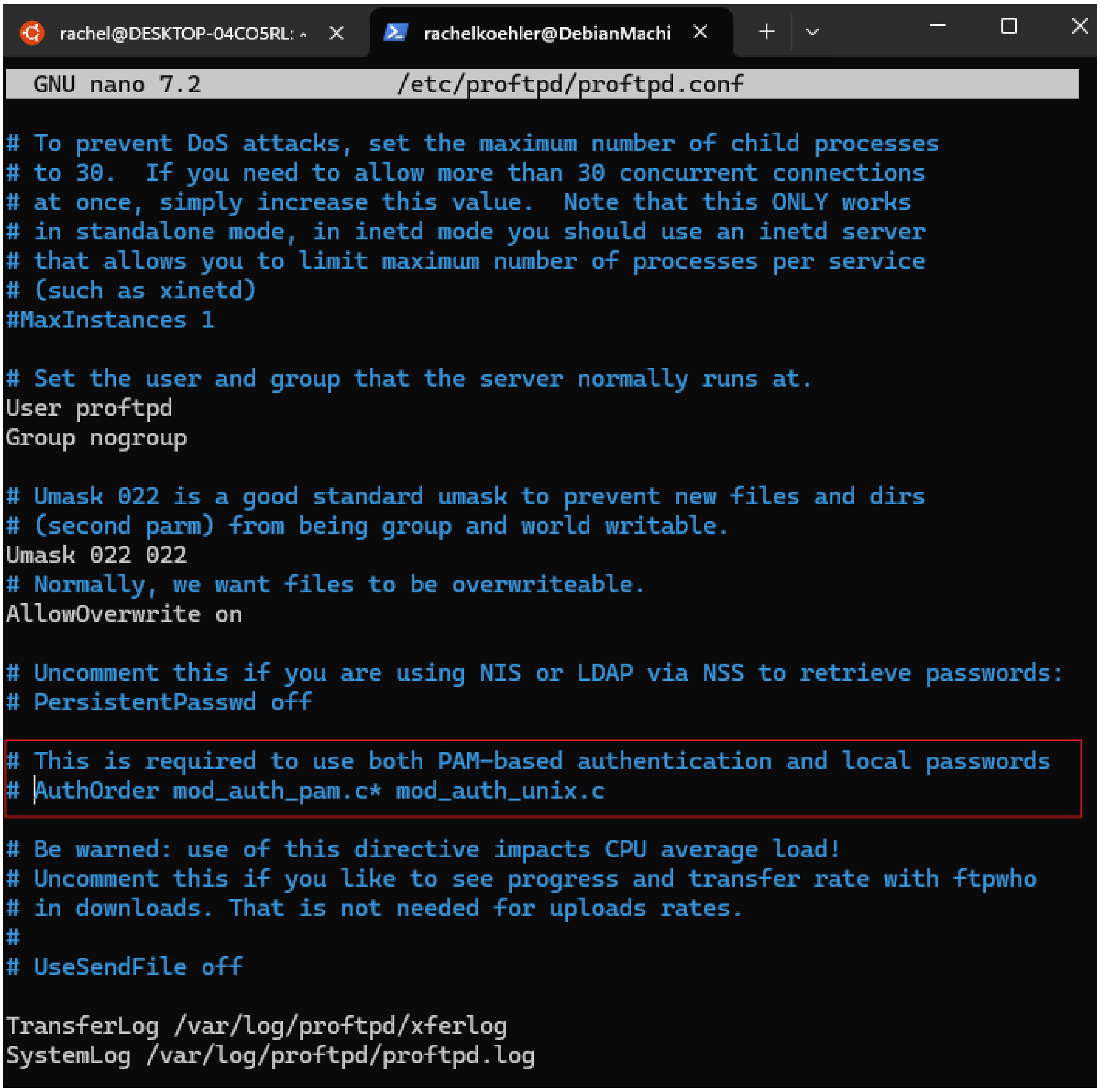


```
proftpd-core                  1.3.8+dfsg-4+deb12u4          amd64        Versatile, virtual-hosting FTP daemon - binaries
proftpd-doc                   1.3.8+dfsg-4+deb12u4          all          Versatile, virtual-hosting FTP daemon - documentation
```

Enable LDAP Authentication in ProFTPD

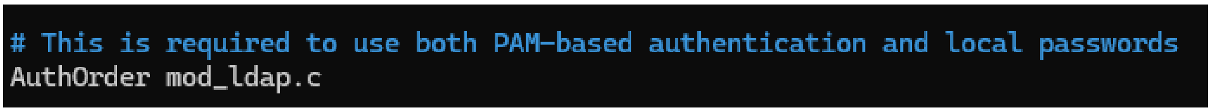


```
rachelkoehler@DebianMachine1:~$ sudo nano /etc/proftpd/proftpd.conf
```



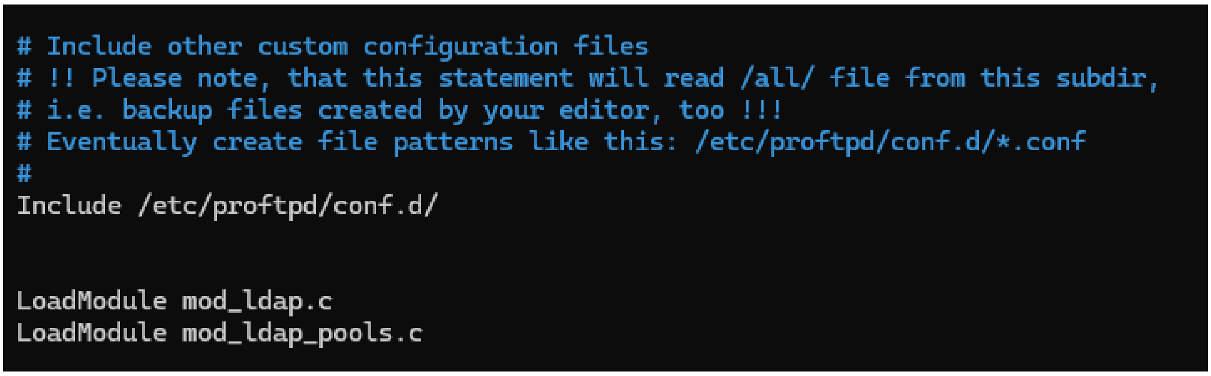
```
  GNU nano 7.2                 /etc/proftpd/proftpd.conf

# To prevent DoS attacks, set the maximum number of child processes
# to 30.  If you need to allow more than 30 concurrent connections
# at once, simply increase this value.  Note that this ONLY works
# in standalone mode, in inetd mode you should use an inetd server
# that allows you to limit maximum number of processes per service
# (such as xinetd)
#MaxInstances 1

# Set the user and group that the server normally runs at.
User proftpd
Group nogroup

# Umask 022 is a good standard umask to prevent new files and dirs
# (second parm) from being group and world writable.
Umask 022 022
# Normally, we want files to be overwriteable.
AllowOverwrite on

# Uncomment this if you are using NIS or LDAP via NSS to retrieve passwords:
# PersistentPasswd off

# This is required to use both PAM-based authentication and local passwords
# AuthOrder mod_auth_pam.c* mod_auth_unix.c

# Be warned: use of this directive impacts CPU average load!
# Uncomment this if you like to see progress and transfer rate with ftpwho
# in downloads. That is not needed for uploads rates.
#
# UseSendFile off

TransferLog /var/log/proftpd/xferlog
SystemLog /var/log/proftpd/proftpd.log
```

Devient…



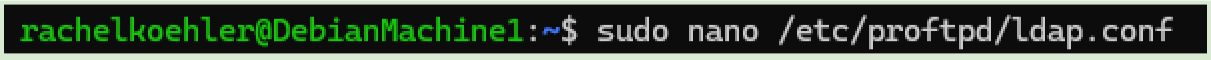
```
# This is required to use both PAM-based authentication and local passwords
AuthOrder mod_ldap.c
```

Rajouté à la fin pour permettre l'authentification LDAP



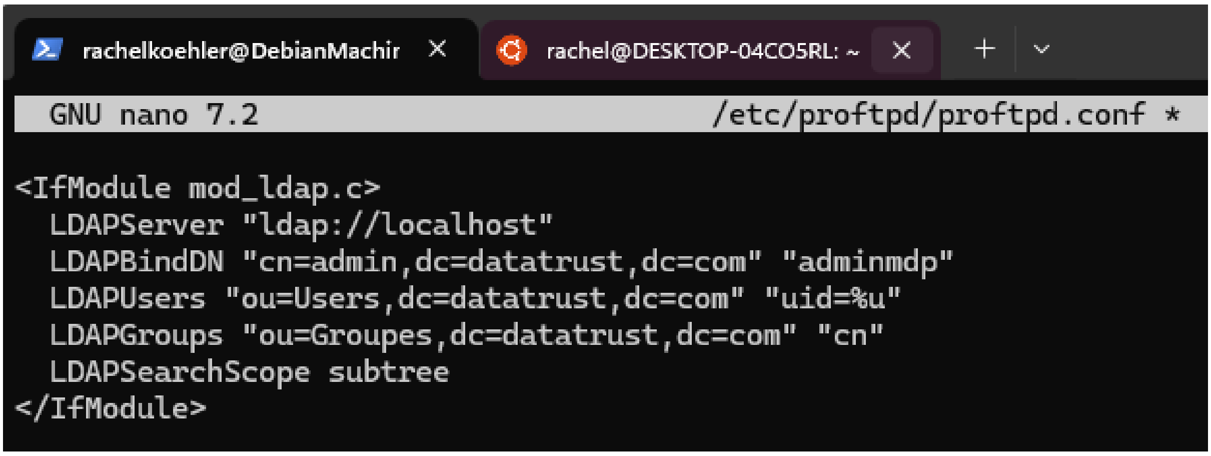
```
# Include other custom configuration files
# !! Please note, that this statement will read /all/ file from this subdir,
# i.e. backup files created by your editor, too !!!
# Eventually create file patterns like this: /etc/proftpd/conf.d/*.conf
#
Include /etc/proftpd/conf.d/


LoadModule mod_ldap.c
LoadModule mod_ldap_pools.c
```

Configure LDAP Authentication

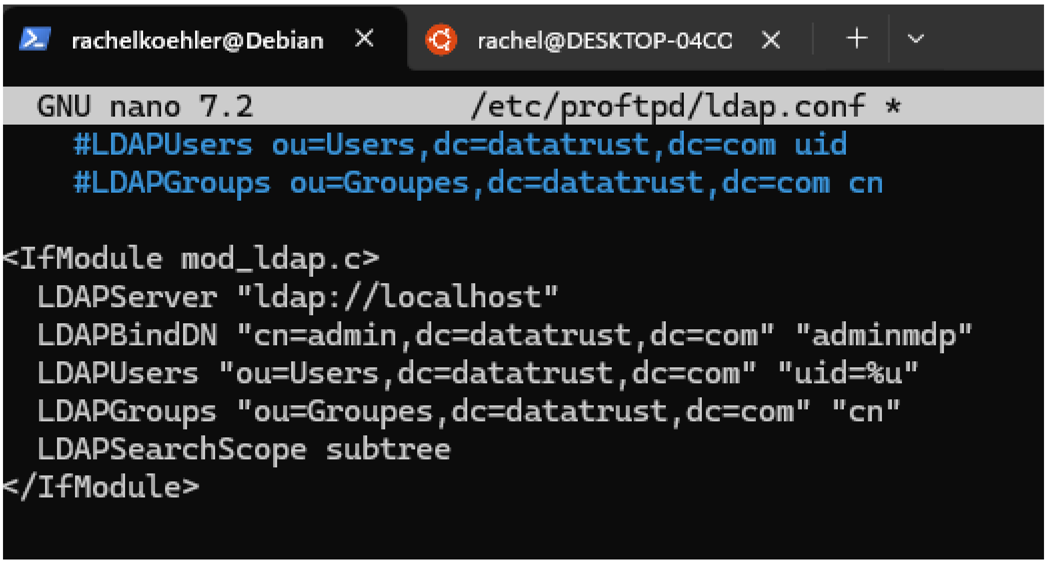


```
rachelkoehler@DebianMachine1:~$ sudo nano /etc/proftpd/ldap.conf
```

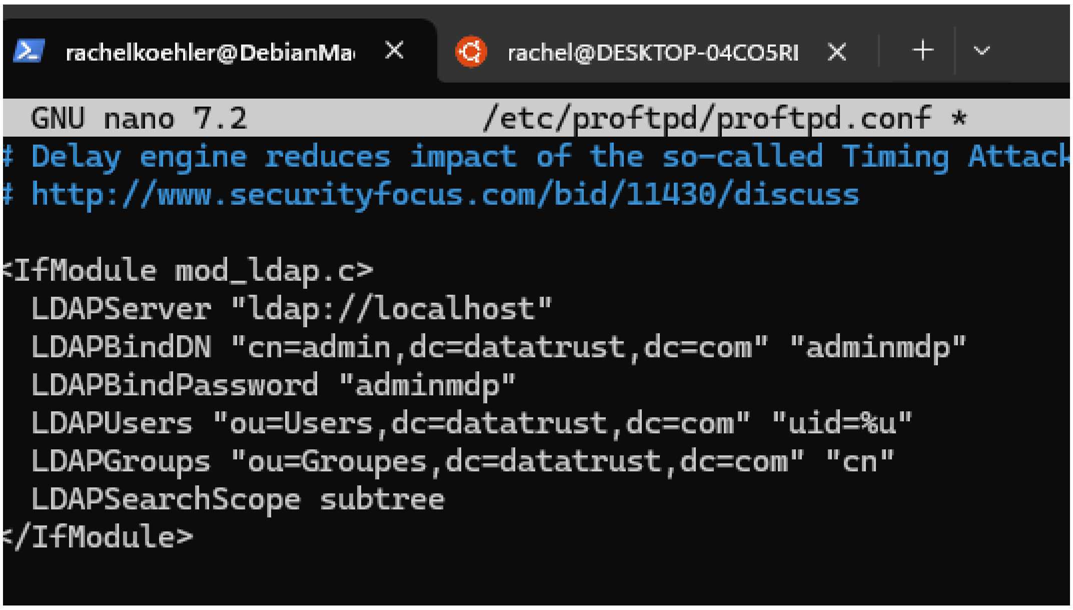
Ajouter script:



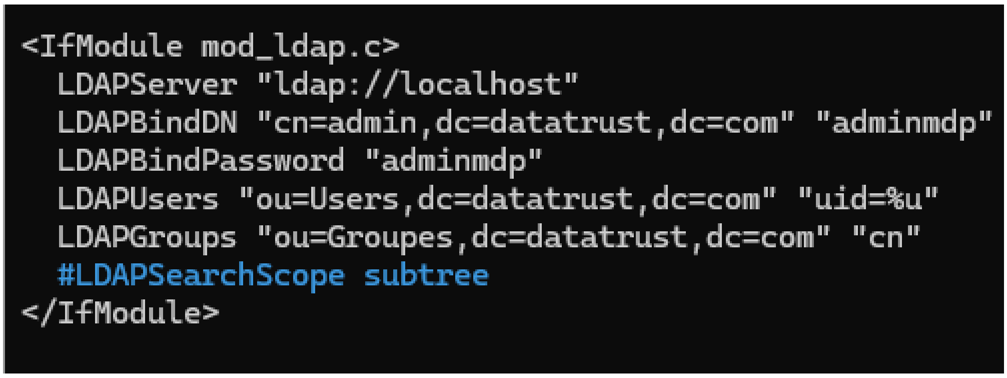
```
  GNU nano 7.2                              /etc/proftpd/proftpd.conf *

<IfModule mod_ldap.c>
  LDAPServer "ldap://localhost"
  LDAPBindDN "cn=admin,dc=datatrust,dc=com" "adminmdp"
  LDAPUsers "ou=Users,dc=datatrust,dc=com" "uid=%u"
  LDAPGroups "ou=Groupes,dc=datatrust,dc=com" "cn"
  LDAPSearchScope subtree
</IfModule>
```



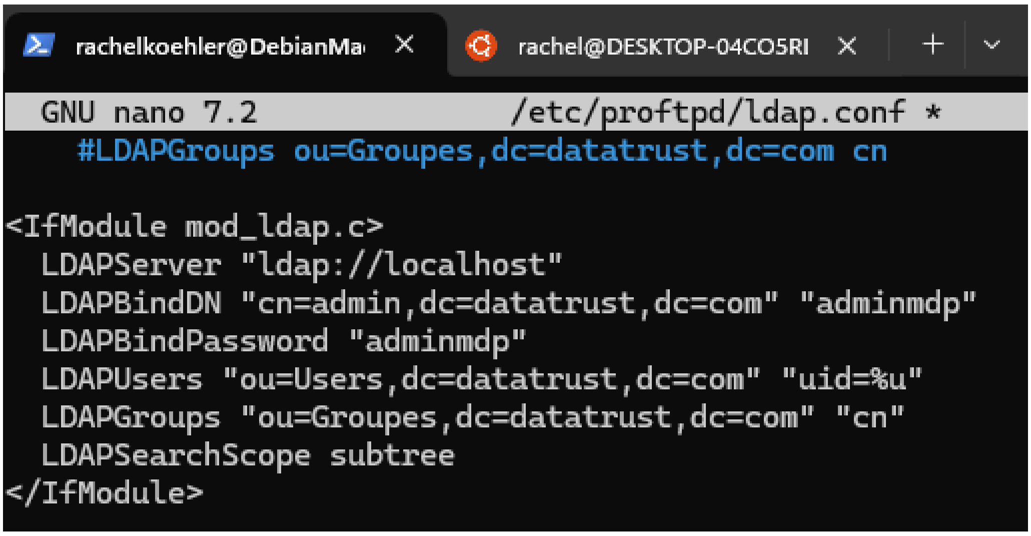
```
 GNU nano 7.2                /etc/proftpd/ldap.conf *
    #LDAPUsers ou=Users,dc=datatrust,dc=com uid
    #LDAPGroups ou=Groupes,dc=datatrust,dc=com cn

<IfModule mod_ldap.c>
  LDAPServer "ldap://localhost"
  LDAPBindDN "cn=admin,dc=datatrust,dc=com" "adminmdp"
  LDAPUsers "ou=Users,dc=datatrust,dc=com" "uid=%u"
  LDAPGroups "ou=Groupes,dc=datatrust,dc=com" "cn"
  LDAPSearchScope subtree
</IfModule>
```



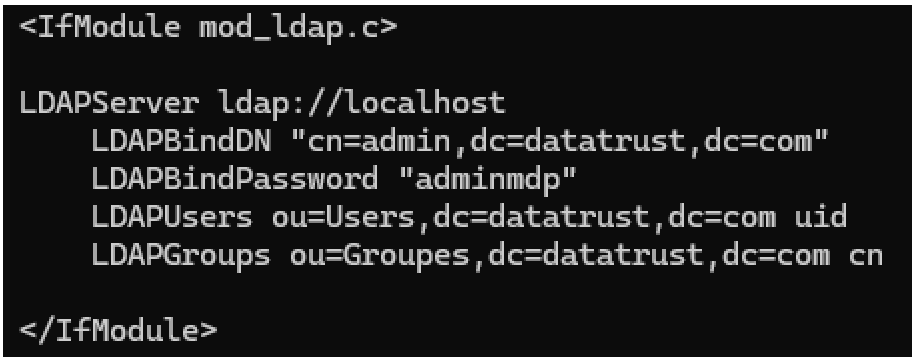
```
 GNU nano 7.2                /etc/proftpd/proftpd.conf *
# Delay engine reduces impact of the so-called Timing Attack
# http://www.securityfocus.com/bid/11430/discuss

<IfModule mod_ldap.c>
  LDAPServer "ldap://localhost"
  LDAPBindDN "cn=admin,dc=datatrust,dc=com" "adminmdp"
  LDAPBindPassword "adminmdp"
  LDAPUsers "ou=Users,dc=datatrust,dc=com" "uid=%u"
  LDAPGroups "ou=Groupes,dc=datatrust,dc=com" "cn"
  LDAPSearchScope subtree
</IfModule>
```



```
<IfModule mod_ldap.c>
  LDAPServer "ldap://localhost"
  LDAPBindDN "cn=admin,dc=datatrust,dc=com" "adminmdp"
  LDAPBindPassword "adminmdp"
  LDAPUsers "ou=Users,dc=datatrust,dc=com" "uid=%u"
  LDAPGroups "ou=Groupes,dc=datatrust,dc=com" "cn"
  #LDAPSearchScope subtree
</IfModule>
```

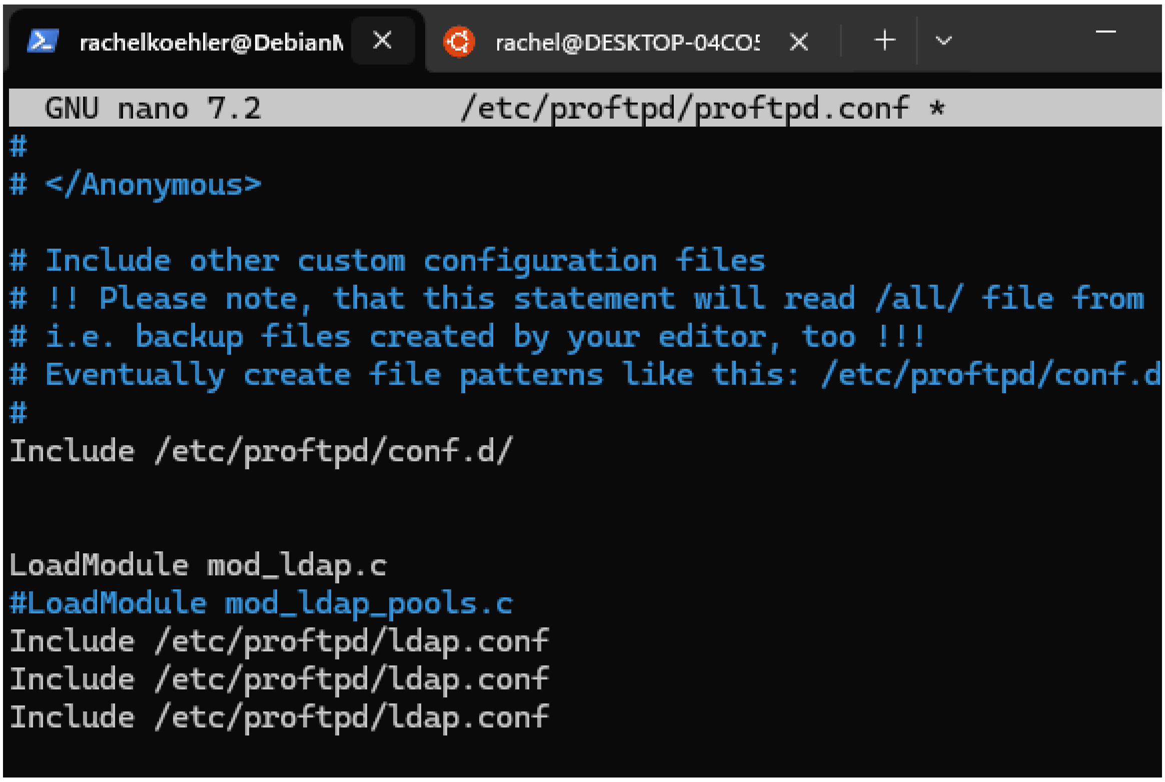


```
 GNU nano 7.2                 /etc/proftpd/ldap.conf *
    #LDAPGroups ou=Groupes,dc=datatrust,dc=com cn

<IfModule mod_ldap.c>
  LDAPServer "ldap://localhost"
  LDAPBindDN "cn=admin,dc=datatrust,dc=com" "adminmdp"
  LDAPBindPassword "adminmdp"
  LDAPUsers "ou=Users,dc=datatrust,dc=com" "uid=%u"
  LDAPGroups "ou=Groupes,dc=datatrust,dc=com" "cn"
  LDAPSearchScope subtree
</IfModule>
```



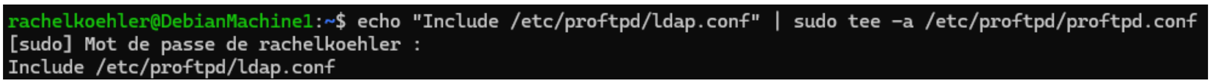
```
<IfModule mod_ldap.c>

LDAPServer ldap://localhost
    LDAPBindDN "cn=admin,dc=datatrust,dc=com"
    LDAPBindPassword "adminmdp"
    LDAPUsers ou=Users,dc=datatrust,dc=com uid
    LDAPGroups ou=Groupes,dc=datatrust,dc=com cn

</IfModule>
```

Besoin de commenter ligne (si existante dns le fichier de conf)



```
  GNU nano 7.2              /etc/proftpd/proftpd.conf *
#
# </Anonymous>

# Include other custom configuration files
# !! Please note, that this statement will read /all/ file from
# i.e. backup files created by your editor, too !!!
# Eventually create file patterns like this: /etc/proftpd/conf.d
#
Include /etc/proftpd/conf.d/


LoadModule mod_ldap.c
#LoadModule mod_ldap_pools.c
Include /etc/proftpd/ldap.conf
Include /etc/proftpd/ldap.conf
Include /etc/proftpd/ldap.conf
```

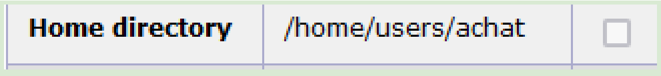
Now link this configuration in **ProFTPD**:



```
rachelkoehler@DebianMachine1:~$ echo "Include /etc/proftpd/ldap.conf" | sudo tee -a /etc/proftpd/proftpd.conf
[sudo] Mot de passe de rachelkoehler :
Include /etc/proftpd/ldap.conf
```

Set Up Home Directories for FTP Users

Déjà configuré en amont 👍

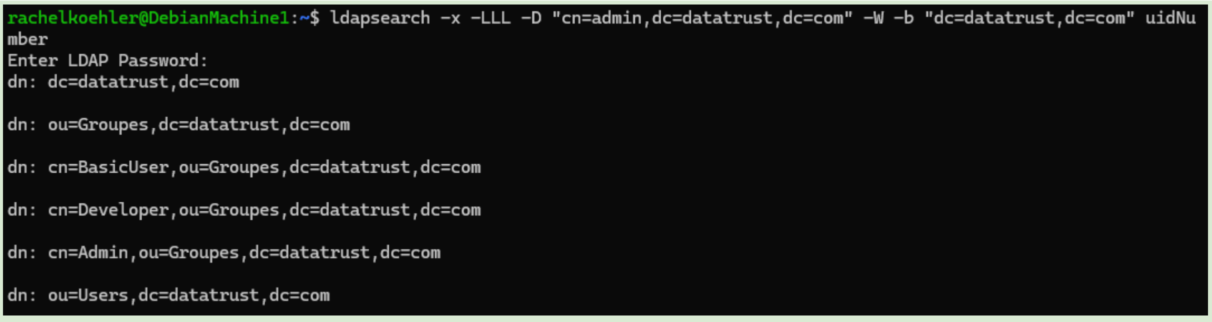


Home directory	/home/users/achat	□

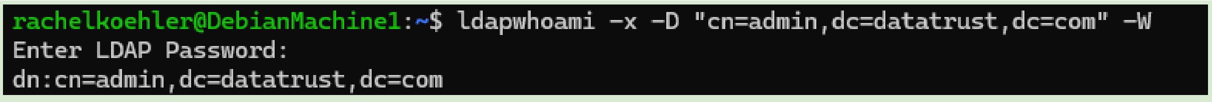
How to Test LDAP Authentication with the Admin Account

If you still want to test authentication using the LDAP administrator, follow these steps:

1️⃣Check if the Admin User Has a UID

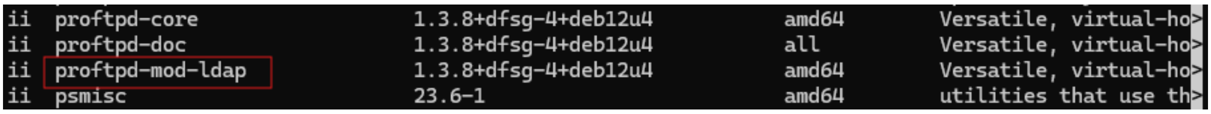


```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -LLL -D "cn=admin,dc=datatrust,dc=com" -W -b "dc=datatrust,dc=com" uidNu
mber
Enter LDAP Password:
dn: dc=datatrust,dc=com

dn: ou=Groupes,dc=datatrust,dc=com

dn: cn=BasicUser,ou=Groupes,dc=datatrust,dc=com

dn: cn=Developer,ou=Groupes,dc=datatrust,dc=com

dn: cn=Admin,ou=Groupes,dc=datatrust,dc=com

dn: ou=Users,dc=datatrust,dc=com
```

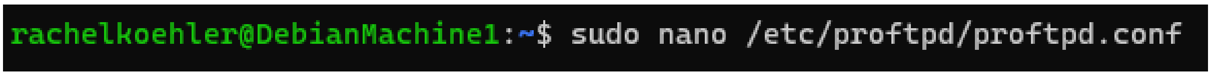


```
rachelkoehler@DebianMachine1:~$ ldapwhoami -x -D "cn=admin,dc=datatrust,dc=com" -W
Enter LDAP Password:
dn:cn=admin,dc=datatrust,dc=com
```

Nouveau tutoriel FTP x LDAP à suivre

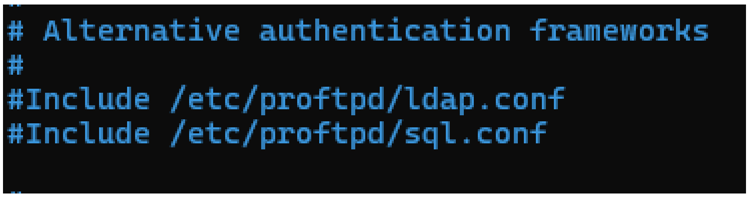


```
ii  proftpd-core                 1.3.8+dfsg-4+deb12u4       amd64        Versatile, virtual-ho>
ii  proftpd-doc                  1.3.8+dfsg-4+deb12u4       all          Versatile, virtual-ho>
ii  proftpd-mod-ldap             1.3.8+dfsg-4+deb12u4       amd64        Versatile, virtual-ho>
ii  psmisc                       23.6-1                     amd64        utilities that use th>
```

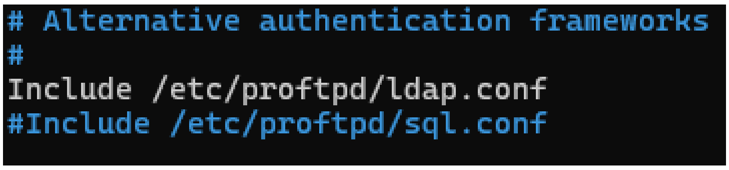


```
rachelkoehler@DebianMachine1:~$ sudo nano /etc/proftpd/proftpd.conf
```

On décommente

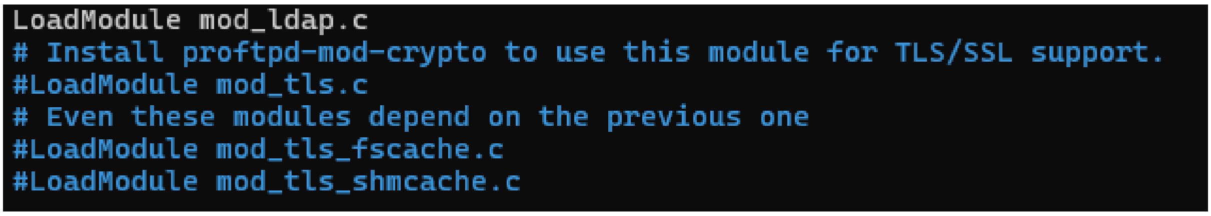


```
# Alternative authentication frameworks
#
#Include /etc/proftpd/ldap.conf
#Include /etc/proftpd/sql.conf
```

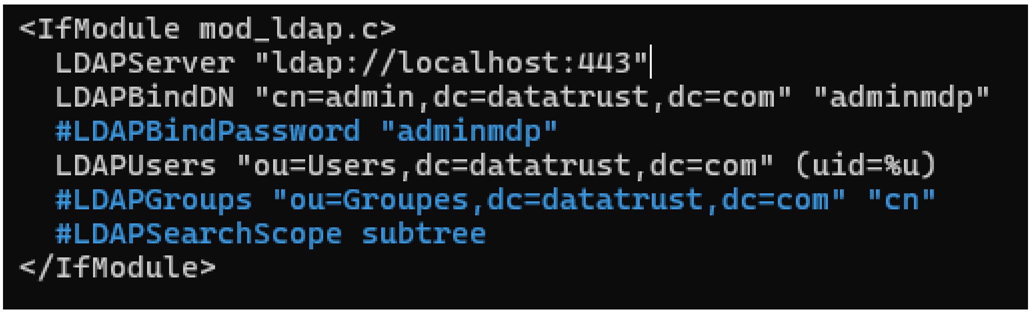


```
# Alternative authentication frameworks
#
Include /etc/proftpd/ldap.conf
#Include /etc/proftpd/sql.conf
```

Décommenter LoadModule mod_ldap.c

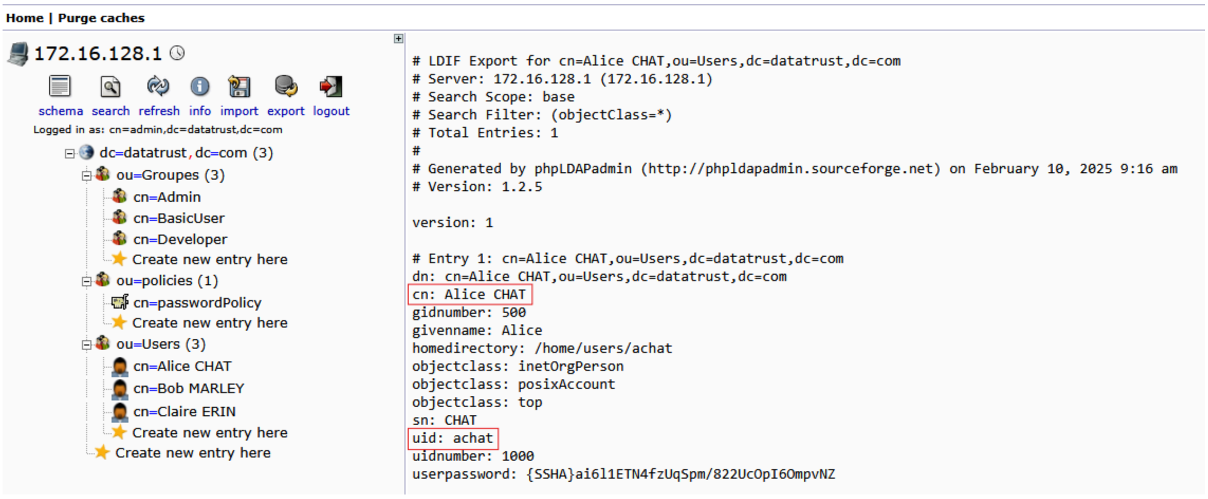


```
LoadModule mod_ldap.c
# Install proftpd-mod-crypto to use this module for TLS/SSL support.
#LoadModule mod_tls.c
# Even these modules depend on the previous one
#LoadModule mod_tls_fscache.c
#LoadModule mod_tls_shmcache.c
```

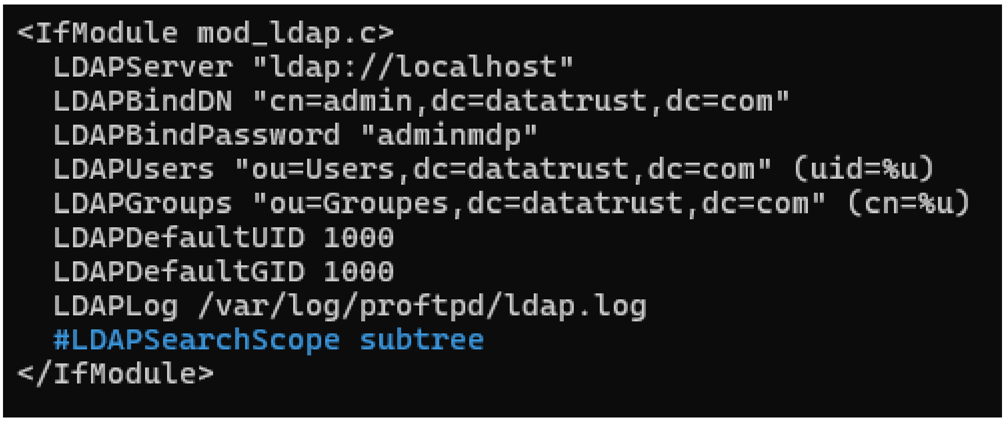


```
<IfModule mod_ldap.c>
  LDAPServer "ldap://localhost:443"
  LDAPBindDN "cn=admin,dc=datatrust,dc=com" "adminmdp"
  #LDAPBindPassword "adminmdp"
  LDAPUsers "ou=Users,dc=datatrust,dc=com" (uid=%u)
  #LDAPGroups "ou=Groupes,dc=datatrust,dc=com" "cn"
  #LDAPSearchScope subtree
</IfModule>
```

NOUVEAU TUTORIEL



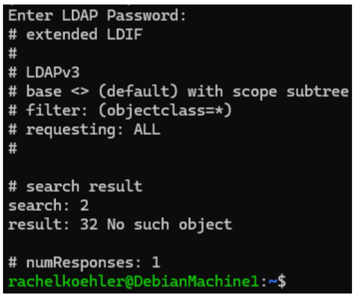
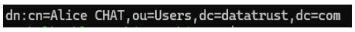
```
# LDIF Export for cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
# Server: 172.16.128.1 (172.16.128.1)
# Search Scope: base
# Search Filter: (objectClass=*)
# Total Entries: 1
#
# Generated by phpLDAPadmin (http://phpldapadmin.sourceforge.net) on February 10, 2025 9:16 am
# Version: 1.2.5

version: 1

# Entry 1: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
dn: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
cn: Alice CHAT
gidnumber: 500
givenname: Alice
homedirectory: /home/users/achat
objectclass: inetOrgPerson
objectclass: posixAccount
objectclass: top
sn: CHAT
uid: achat
uidnumber: 1000
userpassword: {SSHA}ai6l1ETN4fzUqSpm/822UcOpI6OmpvNZ
```



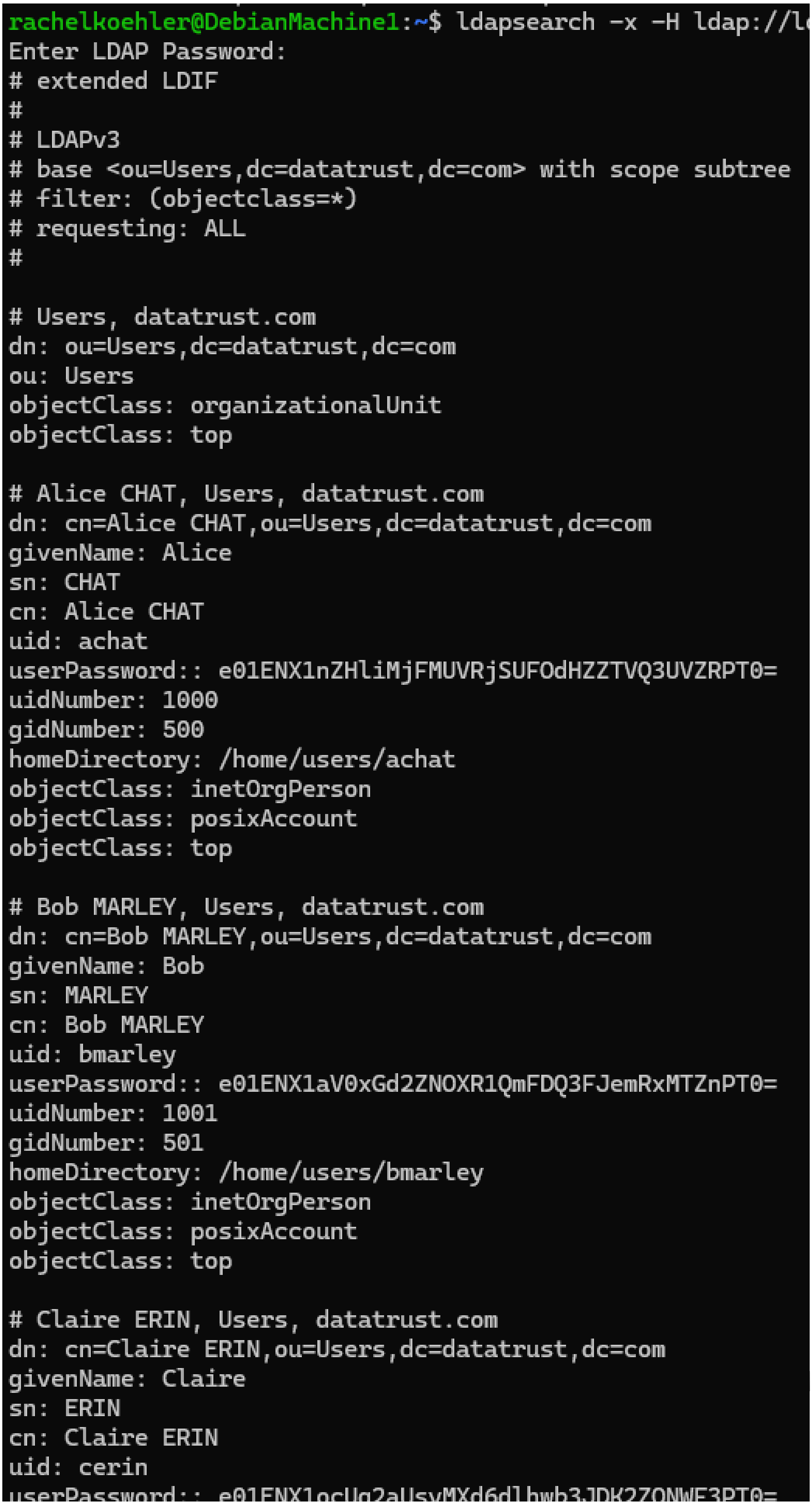
```
<IfModule mod_ldap.c>
  LDAPServer "ldap://localhost"
  LDAPBindDN "cn=admin,dc=datatrust,dc=com"
  LDAPBindPassword "adminmdp"
  LDAPUsers "ou=Users,dc=datatrust,dc=com" (uid=%u)
  LDAPGroups "ou=Groupes,dc=datatrust,dc=com" (cn=%u)
  LDAPDefaultUID 1000
  LDAPDefaultGID 1000
  LDAPLog /var/log/proftpd/ldap.log
  #LDAPSearchScope subtree
</IfModule>
```

CAS DE FIGURE RENCONTRÉS

S’agissant des tests d’accès

ALICE - info	***Commande rentrée***	***Erreur rencontrée***
dn: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com givenName: Alice sn: CHAT cn: Alice CHAT uid: achat userPassword:: e01ENX1nZHliMjFMUVRjSUFOdHZZTVQ3UVZRPT0= uidNumber: 1000 gidNumber: 500 homeDirectory: /home/users/achat objectClass: inetOrgPerson objectClass: posixAccount objectClass: top	ldapsearch -x -H ldap://localhost -D "cn=Alice CHAT,ou=Users ,dc=datatrust,dc=com" -W	 Enter LDAP Password: # extended LDIF # # LDAPv3 # base <> (default) with scope subtree # filter: (objectclass=*) # requesting: ALL # # search result search: 2 result: 32 No such object # numResponses: 1 rachelkoehler@DebianMachine1:~$ ***This means that LDAP cannot find the specified***
	ldapwhoami -x -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com" -W	 dn:cn=Alice CHAT,ou=Users,dc=datatrust,dc=com ***This means that it works***
She may need some access/permissions car la command marche pour l’Admin:		

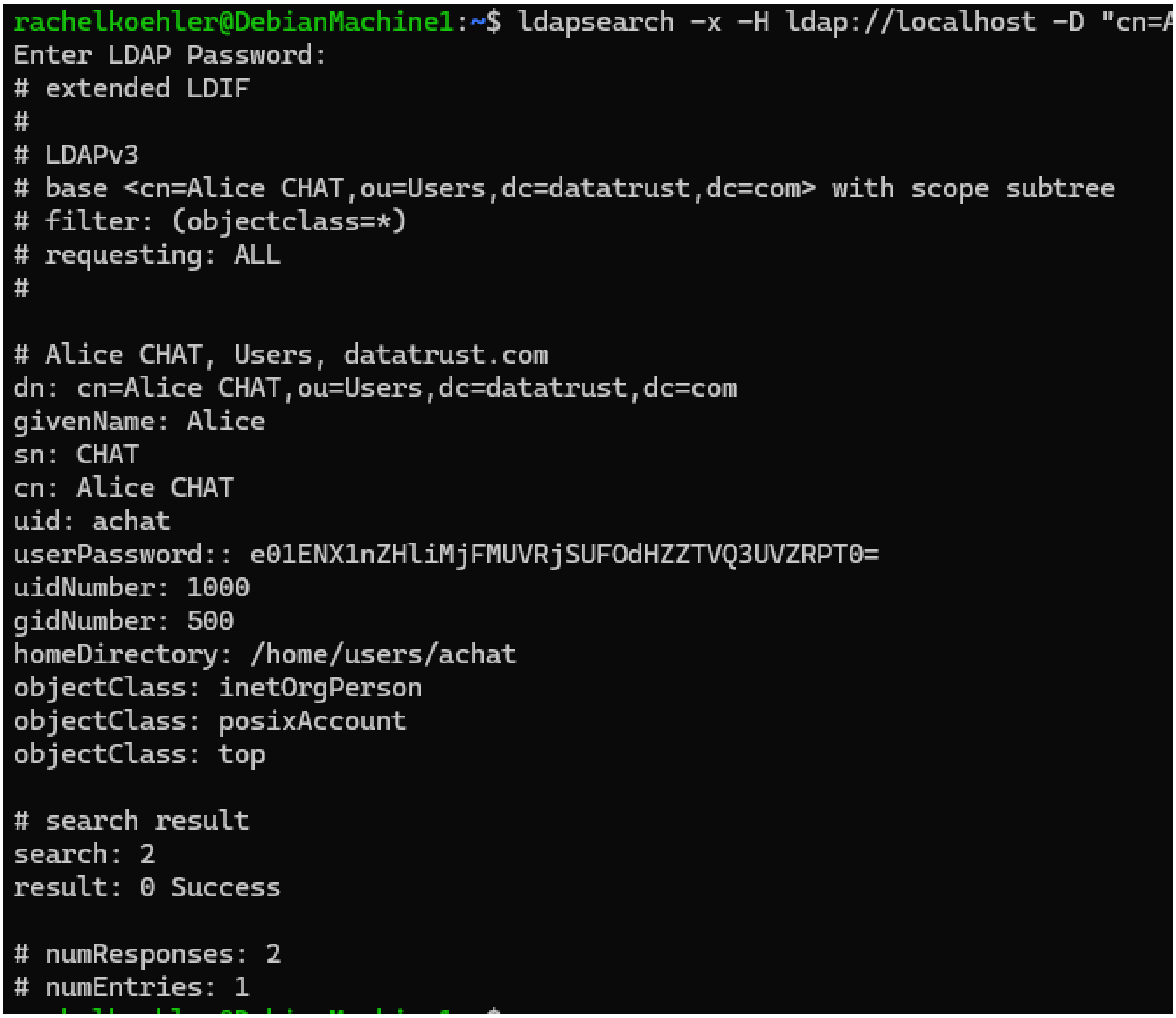
```
ldapsearch -x -H ldap://localhost -D "cn=admin,dc=datatrust,dc=com" -W -b
"ou=Users,dc=datatrust,dc=com"
```



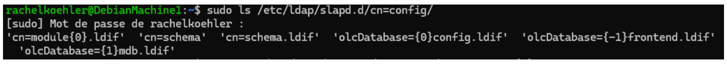
```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://l
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <ou=Users,dc=datatrust,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# Users, datatrust.com
dn: ou=Users,dc=datatrust,dc=com
ou: Users
objectClass: organizationalUnit
objectClass: top

# Alice CHAT, Users, datatrust.com
dn: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
givenName: Alice
sn: CHAT
cn: Alice CHAT
uid: achat
userPassword:: e01ENX1nZHliMjFMUVRjSUFOdHZZTVQ3UVZRPT0=
uidNumber: 1000
gidNumber: 500
homeDirectory: /home/users/achat
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# Bob MARLEY, Users, datatrust.com
dn: cn=Bob MARLEY,ou=Users,dc=datatrust,dc=com
givenName: Bob
sn: MARLEY
cn: Bob MARLEY
uid: bmarley
userPassword:: e01ENX1aV0xGd2ZNOXR1QmFDQ3FJemRxMTZnPT0=
uidNumber: 1001
gidNumber: 501
homeDirectory: /home/users/bmarley
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# Claire ERIN, Users, datatrust.com
dn: cn=Claire ERIN,ou=Users,dc=datatrust,dc=com
givenName: Claire
sn: ERIN
cn: Claire ERIN
uid: cerin
userPassword:: e01ENX1ocUg2aUsvMXd6dlhwb3JDK2ZONWE3PT0=
```


Aussi elle est capable de lire son propre fichier ldif, via la commande
ldapsearch -x -H ldap://localhost -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com" -W -b "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com"

Donc c'est définitivement un pb de permissions

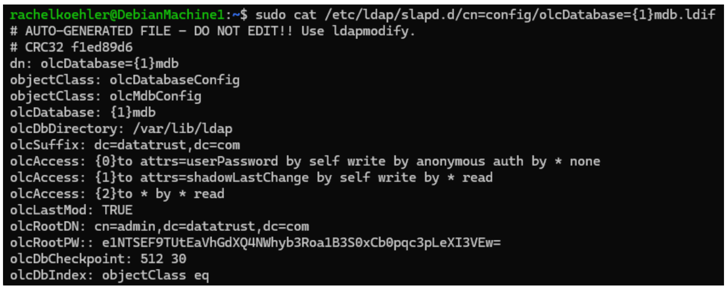


```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://localhost -D "cn=A
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <cn=Alice CHAT,ou=Users,dc=datatrust,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# Alice CHAT, Users, datatrust.com
dn: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
givenName: Alice
sn: CHAT
cn: Alice CHAT
uid: achat
userPassword:: e01ENX1nZHliMjFMUVRjSUFOdHZZTVQ3UVZRPT0=
uidNumber: 1000
gidNumber: 500
homeDirectory: /home/users/achat
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```



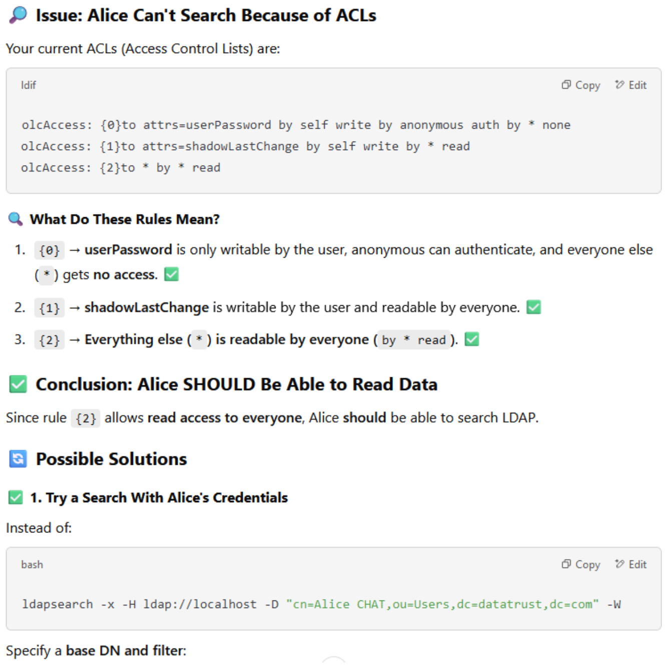
```
rachelkoehler@DebianMachine1:~$ sudo ls /etc/ldap/slapd.d/cn=config/
[sudo] Mot de passe de rachelkoehler :
'cn=module{0}.ldif'  'cn=schema'  'cn=schema.ldif'  'olcDatabase={0}config.ldif'  'olcDatabase={-1}frontend.ldif'
  'olcDatabase={1}mdb.ldif'
```

Go in the mdb.ldif file



```
rachelkoehler@DebianMachine1:~$ sudo cat /etc/ldap/slapd.d/cn=config/olcDatabase={1}mdb.ldif
# AUTO-GENERATED FILE - DO NOT EDIT!! Use ldapmodify.
# CRC32 f1ed89d6
dn: olcDatabase={1}mdb
objectClass: olcDatabaseConfig
objectClass: olcMdbConfig
olcDatabase: {1}mdb
olcDbDirectory: /var/lib/ldap
olcSuffix: dc=datatrust,dc=com
olcAccess: {0}to attrs=userPassword by self write by anonymous auth by * none
olcAccess: {1}to attrs=shadowLastChange by self write by * read
olcAccess: {2}to * by * read
olcLastMod: TRUE
olcRootDN: cn=admin,dc=datatrust,dc=com
olcRootPW:: e1NTSEF9TUtEaVhGdXQ4NWhyb3Roa1B3S0xCb0pqc3pLeXI3VEw=
olcDbCheckpoint: 512 30
olcDbIndex: objectClass eq
```

SOLUTION



🔎 Issue: Alice Can't Search Because of ACLs

Your current ACLs (Access Control Lists) are:

```ldif
olcAccess: {0}to attrs=userPassword by self write by anonymous auth by * none
olcAccess: {1}to attrs=shadowLastChange by self write by * read
olcAccess: {2}to * by * read
```

🔍 What Do These Rules Mean?

1. `{0}` → **userPassword** is only writable by the user, anonymous can authenticate, and everyone else (`*`) gets **no access**. ✅
2. `{1}` → **shadowLastChange** is writable by the user and readable by everyone. ✅
3. `{2}` → **Everything else** (`*`) **is readable by everyone** (`by * read`). ✅

✅ Conclusion: Alice SHOULD Be Able to Read Data

Since rule `{2}` allows **read access to everyone**, Alice **should** be able to search LDAP.

🔄 Possible Solutions

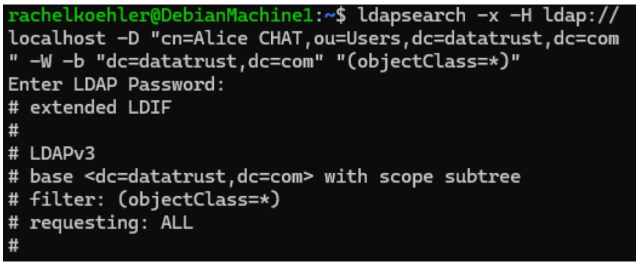
✅ 1. Try a Search With Alice's Credentials

Instead of:

```bash
ldapsearch -x -H ldap://localhost -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com" -W
```

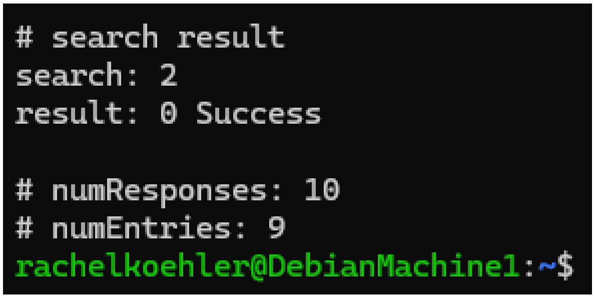
Specify a **base DN** and **filter**:

ldapsearch -x -H ldap://localhost -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com" -W -b "dc=datatrust,dc=com" "(objectClass=*)"



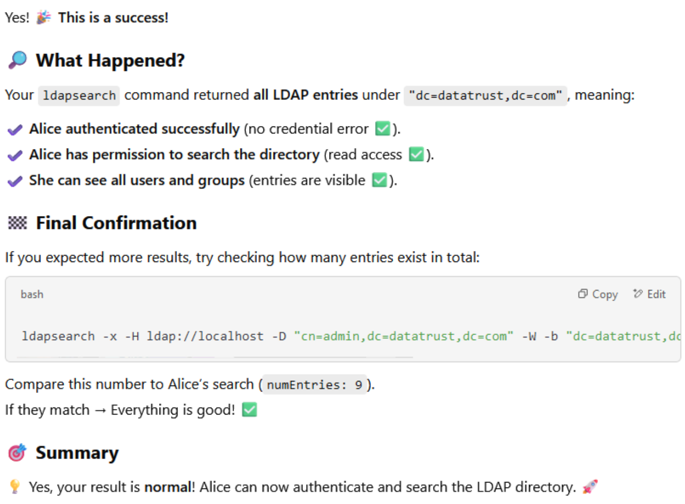
```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://
localhost -D "cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
" -W -b "dc=datatrust,dc=com" "(objectClass=*)"
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=datatrust,dc=com> with scope subtree
# filter: (objectClass=*)
# requesting: ALL
#
```

(...)



```
# search result
search: 2
result: 0 Success

# numResponses: 10
# numEntries: 9
rachelkoehler@DebianMachine1:~$
```

SUCCESS !!!!



Yes! 🎉 **This is a success!**

🔎 What Happened?

Your `ldapsearch` command returned **all LDAP entries** under `"dc=datatrust,dc=com"`, meaning:

✔ **Alice authenticated successfully** (no credential error ✅).
✔ **Alice has permission to search the directory** (read access ✅).
✔ **She can see all users and groups** (entries are visible ✅).

🏁 Final Confirmation

If you expected more results, try checking how many entries exist in total:

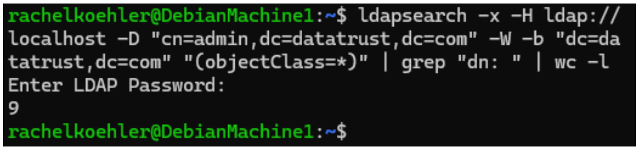
```bash
ldapsearch -x -H ldap://localhost -D "cn=admin,dc=datatrust,dc=com" -W -b "dc=datatrust,dc
```

Compare this number to Alice's search (`numEntries: 9`).
If they match → Everything is good! ✅

🎯 Summary

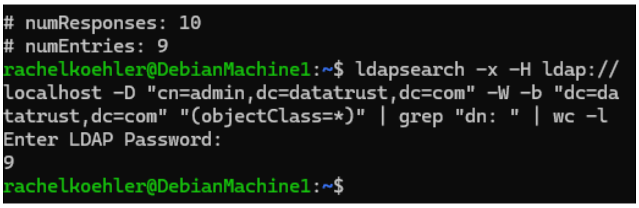
💡 Yes, your result is **normal!** Alice can now authenticate and search the LDAP directory. 🚀

Compare to Alice search (from screen above):



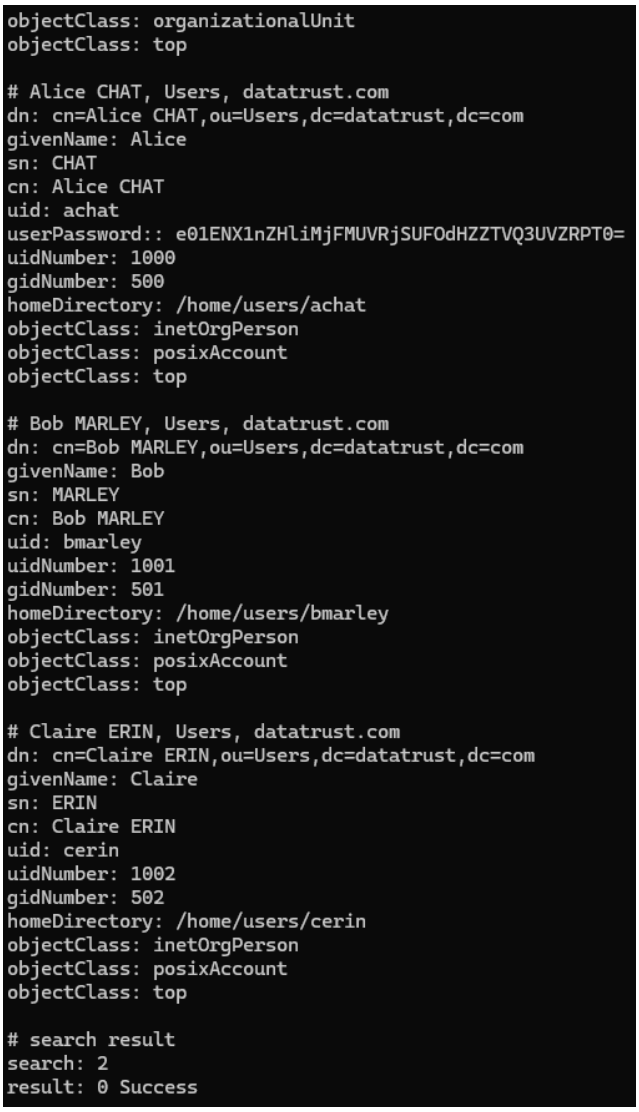
```
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://
localhost -D "cn=admin,dc=datatrust,dc=com" -W -b "dc=da
tatrust,dc=com" "(objectClass=*)" | grep "dn: " | wc -l
Enter LDAP Password:
9
rachelkoehler@DebianMachine1:~$
```

or (well it's the same)

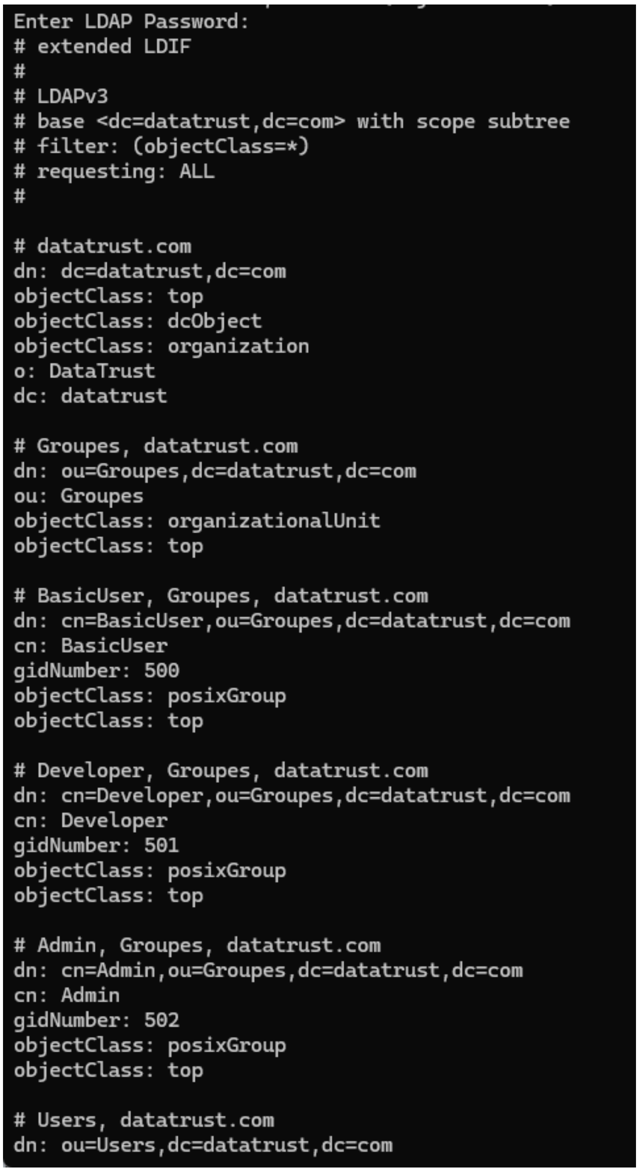


```
# numResponses: 10
# numEntries: 9
rachelkoehler@DebianMachine1:~$ ldapsearch -x -H ldap://
localhost -D "cn=admin,dc=datatrust,dc=com" -W -b "dc=da
tatrust,dc=com" "(objectClass=*)" | grep "dn: " | wc -l
Enter LDAP Password:
9
rachelkoehler@DebianMachine1:~$
```

Jean-François (Jeff) **ANDRIATSIMEVA** - Moussa **CAMARA** Rachel **KOEHLER** - Thomas **VERAN**



```
objectClass: organizationalUnit
objectClass: top

# Alice CHAT, Users, datatrust.com
dn: cn=Alice CHAT,ou=Users,dc=datatrust,dc=com
givenName: Alice
sn: CHAT
cn: Alice CHAT
uid: achat
userPassword:: e01ENX1nZHliMjFMUVRjSUFOdHZZTVQ3UVZRPT0=
uidNumber: 1000
gidNumber: 500
homeDirectory: /home/users/achat
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# Bob MARLEY, Users, datatrust.com
dn: cn=Bob MARLEY,ou=Users,dc=datatrust,dc=com
givenName: Bob
sn: MARLEY
cn: Bob MARLEY
uid: bmarley
uidNumber: 1001
gidNumber: 501
homeDirectory: /home/users/bmarley
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# Claire ERIN, Users, datatrust.com
dn: cn=Claire ERIN,ou=Users,dc=datatrust,dc=com
givenName: Claire
sn: ERIN
cn: Claire ERIN
uid: cerin
uidNumber: 1002
gidNumber: 502
homeDirectory: /home/users/cerin
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top

# search result
search: 2
result: 0 Success
```



```
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=datatrust,dc=com> with scope subtree
# filter: (objectClass=*)
# requesting: ALL
#

# datatrust.com
dn: dc=datatrust,dc=com
objectClass: top
objectClass: dcObject
objectClass: organization
o: DataTrust
dc: datatrust

# Groupes, datatrust.com
dn: ou=Groupes,dc=datatrust,dc=com
ou: Groupes
objectClass: organizationalUnit
objectClass: top

# BasicUser, Groupes, datatrust.com
dn: cn=BasicUser,ou=Groupes,dc=datatrust,dc=com
cn: BasicUser
gidNumber: 500
objectClass: posixGroup
objectClass: top

# Developer, Groupes, datatrust.com
dn: cn=Developer,ou=Groupes,dc=datatrust,dc=com
cn: Developer
gidNumber: 501
objectClass: posixGroup
objectClass: top

# Admin, Groupes, datatrust.com
dn: cn=Admin,ou=Groupes,dc=datatrust,dc=com
cn: Admin
gidNumber: 502
objectClass: posixGroup
objectClass: top

# Users, datatrust.com
dn: ou=Users,dc=datatrust,dc=com
```