Configurer un serveur LDAP

Pré-requis. Ce sujet nécessite les deux sujets suivants :
- Postes clients
- Certificat autosigné

1 Introduction

Pourquoi un serveur LDAP ? Dans un réseau structuré de machines, nous avons souvent besoin de partager des informations entre un serveur et des clients. Le serveur DHCP se charge des aspects réseau, mais il nous faut un autre service afin de partager les fichiers `/etc/passwd`, `/etc/shadow`, `/etc/group`, `/etc/hosts`, `/etc/services` et d'autres. C'est l'objet d'un serveur d'annuaire (**directory service**). Il en existe plusieurs et nous allons utiliser un mécanisme basé sur un serveur LDAP (Lightweight Directory Access Protocol).

Comprendre LDAP. Vous devriez prendre quelques minutes pour lire cette rapide introduction aux annuaires LDAP.

2 Le serveur LDAP

- Le logiciel et sa préparation :

Installation et lancement

```
dnf -y install openldap-servers openldap-clients
systemctl enable --now slapd
```

Modification de mot de passe administrateur

```
## Choisissez un mot de passe simple
PASS="$(slappasswd -h '{SSHA}')"
echo "Password is $PASS"

## Construction d'un fichier LDIF afin de changer
## le mot de passe administrateur
cat <<FIN >change-root-password.ldif
dn: olcDatabase={0}config,cn=config
changetype: modify
add: olcRootPW
olcRootPW: $PASS
FIN

## Insertion de ce fichier dans l'annuaire
ldapadd -Y EXTERNAL -H ldapi:/// -f change-root-password.ldif
```

- Ajout des schémas. Un schéma LDAP est une description d'une structure de données (attributs) décrivant des informations que nous allons stocker dans les noeuds LDAP. Ces trois schémas sont utiles pour définir, des utilisateurs, des groupes et des machines.

Ajout des schémas nécessaires

```
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/cosine.ldif
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/nis.ldif
ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/inetorgperson.ldif
```

Note : À partir de cette étape, nous allons choisir notre domaine `dc=idl,dc=xfr` qui va être la racine de notre arbre LDAP. Toutes les autres informations seront placées en dessous.

- L'administrateur peut gérer la totalité de l'annuaire. Un **manager** a des droits sur une partie de l'arbre LDAP. Nous pourrions donc avoir plusieurs arbres gérés par des managers différents dans un même annuaire LDAP. Le manager de notre domaine `dc=idl,dc=xfr` est identifié par le `dn` (distinguished name) `cn=Manager,dc=idl,dc=xfr`.

Donner les droits au manager

```
## Mot de passe du manager (je l'ai fixé, option -s)
PASS="$(slappasswd -h '{SSHA}' -s mhello)"
echo "Password is $PASS"

## Fichier LDIF pour changer le mot de passe manager
## ATTENTION : des lignes sont longues
cat <<FIN >change-domain.ldif
dn: olcDatabase={1}monitor,cn=config
changetype: modify
replace: olcAccess
olcAccess: {0}to * by dn.base="gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth"
  read by dn.base="cn=Manager,dc=idl,dc=xfr" read by * none

dn: olcDatabase={2}mdb,cn=config
changetype: modify
replace: olcSuffix
olcSuffix: dc=idl,dc=xfr

dn: olcDatabase={2}mdb,cn=config
changetype: modify
replace: olcRootDN
olcRootDN: cn=Manager,dc=idl,dc=xfr

dn: olcDatabase={2}mdb,cn=config
changetype: modify
add: olcRootPW
olcRootPW: $PASS

dn: olcDatabase={2}mdb,cn=config
changetype: modify
add: olcAccess
olcAccess: {0}to attrs=userPassword,shadowLastChange by
  dn="cn=Manager,dc=idl,dc=xfr" write by anonymous auth by self write by * none
olcAccess: {1}to dn.base="" by * read
olcAccess: {2}to * by dn="cn=Manager,dc=idl,dc=xfr" write by * read
FIN

## Définir le domaine
ldapmodify -Y EXTERNAL -H ldapi:/// -f change-domain.ldif
```

- Nous allons maintenant créer, avec `ldapadd`, le manager (le noeud) ainsi que deux autres noeuds afin d'organiser les utilisateurs et les groupes.

Création de notre arbre

```
## Toujours un fichier LDIF
cat <<FIN >base-domain.ldif
dn: dc=idl,dc=xfr
objectClass: top
objectClass: dcObject
objectclass: organization
o: IDL domain
dc: idl

dn: cn=Manager,dc=idl,dc=xfr
objectClass: organizationalRole
cn: Manager
description: Directory Manager

dn: ou=People,dc=idl,dc=xfr
objectClass: organizationalUnit
ou: People

dn: ou=Group,dc=idl,dc=xfr
objectClass: organizationalUnit
ou: Group
FIN

## Ajout des noeuds avec authentification du manager
ldapadd -x -D cn=Manager,dc=idl,dc=xfr -w mhello -f base-domain.ldif
```

- Notre arbre est maintenant constitué de quatre noeuds (la racine, le manager et les deux groupes) :

```
dn: dc=idl,dc=xfr
objectClass: top
objectClass: dcObject
objectclass: organization
o: IDL domain
dc: idl
    |
    +--> dn: cn=Manager,dc=idl,dc=xfr
    |    objectClass: organizationalRole
    |    cn: Manager
    |    description: Directory Manager
    |
    +--> dn: ou=People,dc=idl,dc=xfr
    |    objectClass: organizationalUnit
    |    ou: People
    |
    +--> dn: ou=Group,dc=idl,dc=xfr
    |    objectClass: organizationalUnit
    |    ou: Group
```

- Nous pouvons maintenant interroger notre annuaire LDAP (avec `ldapsearch`) :

Trouver des noeuds

```
## La totalité de l'arbre
ldapsearch -x -b "dc=idl,dc=xfr" -H ldap://localhost

## Les noeuds qui possèdent un attribut ou (organizationalUnit)
ldapsearch -x -b "dc=idl,dc=xfr" -H ldap://localhost '(ou=*)'

## Les noeuds pour lesquels ou se termine par 'ple'
ldapsearch -x -b "dc=idl,dc=xfr" -H ldap://localhost '(ou=*ple)'
```

- Nous pouvons même préciser la racine par défaut afin de simplifier nos requêtes :

Configurer la partie client de LDAP

```
cat >> /etc/openldap/ldap.conf <<FIN
BASE  dc=idl,dc=xfr
URI   ldap://srv.idl.xfr
TLS_REQCERT allow
FIN

ldapsearch -x '(ou=*ple)'
```

- Nous allons maintenant déployer notre certificat afin d'utiliser des connexions SSL/TLS :

Configurer SLAPD pour SSL/TLS

```
cp /etc/pki/tls/certs/{server.key,server.crt} /etc/openldap/certs/
chown ldap:ldap /etc/openldap/certs/{server.key,server.crt}

cat <<FIN >mod_ssl.ldif
dn: cn=config
changetype: modify
replace: olcTLSCertificateFile
olcTLSCertificateFile: /etc/openldap/certs/server.crt
-
replace: olcTLSCertificateKeyFile
olcTLSCertificateKeyFile: /etc/openldap/certs/server.key
FIN

ldapmodify -Y EXTERNAL -H ldapi:/// -f mod_ssl.ldif

systemctl restart slapd
```

- Dernière opération de la partie serveur, il nous reste à créer un utilisateur et un groupe :

Ajouter un utilisateur

```
## J'ai fixé le mot de passe à 'uhello'
PASS="$(slappasswd␣-h␣'{SSHA}'␣-s␣uhello)"
echo "Password␣is␣$PASS"

## Fichier LDIF pour l'utilisateur et le groupe
cat <<FIN >ldapuser.ldif
dn: uid=anu01,ou=People,dc=idl,dc=xfr
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
cn: LDAP User
sn: Linux
userPassword: $PASS
loginShell: /bin/bash
uidNumber: 2000
gidNumber: 2000
homeDirectory: /home/anu01

dn: cn=anu01,ou=Group,dc=idl,dc=xfr
objectClass: posixGroup
cn: anu01
gidNumber: 2000
memberUid: anu01
FIN

## Ajouter ces noeuds
ldapadd -x -D cn=Manager,dc=idl,dc=xfr -w mhello -f ldapuser.ldif

## Pour vérifier
ldapsearch -x '(|(uidNumber=*)(gidNumber=*))'
```

- Voila la nouvelle version de notre arbre :

```
dn: dc=idl,dc=xfr
objectClass: top
objectClass: dcObject
objectclass: organization
o: IDL domain
dc: idl
    |
    +--> dn: cn=Manager,dc=idl,dc=xfr
    |    objectClass: organizationalRole
    |    cn: Manager
    |    description: Directory Manager
    |
    +--> dn: ou=People,dc=idl,dc=xfr
    |    objectClass: organizationalUnit
    |    ou: People
    |       |
    |       +--> dn: uid=anu01,ou=People,dc=idl,dc=xfr
    |       |    objectClass: inetOrgPerson
    |       |    objectClass: posixAccount
    |       |    objectClass: shadowAccount
    |       |    cn: LDAP User
    |       |    sn: Linux
    |       |    userPassword: password
    |       |    loginShell: /bin/bash
    |       |    uidNumber: 2000
    |       |    gidNumber: 2000
    |       |    homeDirectory: /home/anu01
    |
    +--> dn: ou=Group,dc=idl,dc=xfr
    |    objectClass: organizationalUnit
    |    ou: Group
    |       |
    |       +--> dn: cn=anu01,ou=Group,dc=idl,dc=xfr
    |       |    objectClass: posixGroup
    |       |    cn: anu01
    |       |    gidNumber: 2000
    |       |    memberUid: anu01
```

3 Configurer un de vos clients

Nous allons maintenant configurer un client afin qu'il puissent récupérer les informations du serveur et ainsi, permettre l'authentification des utilisateurs définis dans l'arbre LDAP.

La partie client de LDAP

```
## Le logiciel nécessaire
dnf -y install openldap-clients sssd sssd-ldap oddjob-mkhomedir

## Changer la stratégie d'authentification
authselect select sssd with-mkhomedir --force

## Configuration de sssd pour lui indiquer l'origine des informations
cat <<FIN >/etc/sssd/sssd.conf
[domain/default]
id_provider = ldap
autofs_provider = ldap
auth_provider = ldap
chpass_provider = ldap
ldap_uri = ldap://srv.idl.xfr/
ldap_search_base = dc=idl,dc=xfr
ldap_id_use_start_tls = True
ldap_tls_cacertdir = /etc/openldap/certs
cache_credentials = True
ldap_tls_reqcert = allow
ldap_library_debug_level = -1

[sssd]
services = nss, pam, autofs
domains = default

[nss]
homedir_substring = /home
FIN

chmod 600 /etc/sssd/sssd.conf

# Redémarrage des services
systemctl restart sssd oddjobd
systemctl enable sssd oddjobd

# Vérifier que l'utilisateur est connu.
id anu01
```

⏩ **Travail à faire :** Essayez de vous connecter par ssh sur le client en utilisant cet utilisateur. Vérifiez que la création de son répertoire d'accueil est bien réalisée (fonctionnalité `mkhomedir`). Vérifiez ensuite (commande `passwd`) que l'utilisateur est en capacité de changer son mot de passe (sur le serveur LDAP).

4 Automatiser la création des utilisateurs

Avec les actions ci-dessous, vous pouvez récupérer un script qui permet de créer dans l'annuaire LDAP, les utilisateurs définis dans le fichier `/etc/passwd` :

```
wget https://jean-luc-massat.pedaweb.univ-amu.fr/ens/cca/passwd2ldap.zip
unzip passwd2ldap.zip
chmod u+x passwd2ldap.sh
```

⏩ **Travail à faire :** Chargez, étudiez et finalement utilisez ce script avec vos anciens utilisateurs.

⏩ **Travail à faire :** Créez un compte classique (avec `useradd`) et synchronisez l'annuaire LDAP.

⏩ **Travail à faire :** Supprimez une entrée LDAP avec `ldapdelete` :

```
ldapdelete -x -D 'cn=Manager,dc=idl,dc=xfr' -w mhello "DN_a_supprimer"
```