OpenLDAP

Olivier Hoarau (olivier.hoarau@funix.org)

V2.0 du 4 août 2022

1 Historique

V2.0 04.08.22 gros toilettage après 6 ans sans mise à jour

V1.6 24.12.16 passage à OpenLDAP 2.4.44 et Admin4 2.2.4, modification pour supprimer des logs et warnings génants

V1.7 15.10.15 passage à la version OpenLDAP 2.4.42, suppression de l'installation de pam_ldap et nss_ldap, présentation de nss_pam_ldapd 0.9.6 et de Admin4 modifications suite passage à Mageia5 et systemd

V1.6 24.12.13

V1.5 04.09.10 passage à la version 2.4.23, suppression de la version rpm

V1.4 05.11.04 passage à la version 2.2.18 et 2.1.25 (RPM mdk)

V1.3 04.05.03 Passage à OpenLDAP 2.1.17 et Mandrake 9.1 (OpenLDAP 2.0.27)

V1.2 24.12.02 Passage à OpenLDAP 2.1.8 et Mandrake 9.0 (OpenLDAP 2.0.25)

V1.1 07.07.02 Passage à OpenLDAP 2.1.2, rajout d'un paragraphe sur l'authentification des utilisateurs du système basé sur LDAP

V1.0 09.06.02 Création du document

2 Préambule

Ce document présente **OpenLDAP** avec une application pratique (authentification des utilisateurs).

La dernière version de ce document est téléchargeable à l'URL https://www.funix.org.

Par ailleurs ce document ne peut pas être utilisé dans un but commercial sans le consentement de son auteur. Ce document vous est fourni "dans l'état" sans aucune garantie de toute sorte, l'auteur ne saurait être tenu responsable des quelconques misères qui pourraient vous arriver lors des manipulations décrites dans ce document.

3 Présentation

LDAP est un protocole basé sur TCP/IP qui permet de partager des bases de données d'information sur un réseau interne (intranet) ou externe (internet). Ces bases de données sont appelées annuaire électronique (Directory en anglais), elles peuvent contenir tout type d'informations, des informations sur les personnes, à des données systèmes. Qui dit base de données, dit recherche, il est donc possible de faire des recherches dans la base en employant plusieurs critères, mais aussi bien sûr de la modifier, mais contrairement à un SGBD, un annuaire est très rapide en lecture, mais l'est beaucoup moins en écriture, en effet comme un annuaire est plutôt lu que modifier il a été optimisé pour la lecture et ne possède pas les mécanismes de transaction complexe que les SGBD possèdent pour traiter de gros volumes de données.
Le **LDAP** ou Lightweight Directory Access Protocol est la version TCP/IP du protocole **DAP**, ce dernier étant le protocole pour accéder au protocole OSI du service d'annuaire X500. Dans un premier temps **LDAP** s'est contenté d'être l'interface à des annuaires X500, mais maintenant **LDAP** peut gérer complètement les bases (standalone **LDAP**).

Si on rentre dans les détails, le protocole **LDAP** est du type client serveur, le serveur contient la base de données, et le client consulte la base de données, le protocole fournit les bases pour cette communication entre la client et le serveur (normalisée par l'IETF par la RFC2251), et les commandes nécessaires au client pour rechercher, créer, modifier ou effacer des données. **LDAP** est bien entendu sécurisé pour le transfert et l'accès aux des données, avec des outils de cryptage comme SSL et d'authentification.

Par ailleurs **LDAP** fournit des outils pour que les serveurs **LDAP** puissent communiquer entre eux, on a ainsi la possibilité de créer des serveurs miroirs qui pourront se synchroniser, ou de relier simplement les serveurs entre eux, les serveurs redirigeant automatiquement les requêtes qui ne les concernent pas.

Les exemples d'applications de **LDAP** sont nombreux:

- bases de données d'employés,
- bases de données de produits,
- bases de données pour certaines applications, exemple :

- toutes les infos contenant les utilisateurs de votre réseau (mot de passe, shell, homedirectory, ...) peuvent être dans la base, on a ainsi beaucoup plus de possibilités qu'un simple fichier **/etc/passwd**, l'authentification peut donc utiliser **LDAP** plutôt que **passwd** ou **shadow** ou encore **NIS**. Vos utilisateurs pourront ainsi changer leur mot de passe et certains de leurs attributs à partir d'une interface web.

- les préférences d'applications ou d'environnement (**netscape**, environnement graphique KDE, ...) sont sauvegardés dans la base, ainsi l'utilisateur peut passer d'une machine à une autre et retrouver ses préférences.

Cette page est une introduction à **LDAP** elle ne couvre pas certains aspects comme les liens avec d'autres bases (duplication, miroir, ...), la sécurité (access control, SSL, ...). Elle n'a seulement pour but de mettre en place un serveur **LDAP** simplement configuré pour que vous puissiez faire vos "premières armes" dans le domaine.

Pour utiliser une base **LDAP** à partir de script **PHP,** voir mon document Apache téléchargeable sur www.funix.org.

4 Format de la base et définitions

4.1 Le Directory Information Tree

Les **LDAP** standalone utilisent le format de base de données **LDBM**, ce dernier utilise le modèle hiérarchique comme le système de fichiers UNIX, c'est à dire qu'il s'apparente à un arbre, qu'on appelle **DIT** (Directory Information Tree). Au sommet de cet arbre se trouve la racine ou suffixe et à chaque nœud de l'arborescence on a un **DSE** (Directory Service Entry) qui correspond à une entrée de l'annuaire. L'entrée située à la racine est appelé **rootDSE** (root Directory Specific Entry), qui décrit la structure de l'arborescence (le **DIT**) ainsi que son contenu.
Chaque entrée est connue de manière unique dans l'arborescence grâce à son **dn** (Distinguished Name). Le **dn** indique le chemin à parcourir pour en partant du sommet arriver à l'entrée correspondante. Par exemple pour identifier une personne, on part du pays (fr), puis le nom de domaine (kervao pour la suite des opérations), le groupe de travail et enfin le nom de la personne, l'ensemble de ces paramètres est le **dn** qui identifie de manière unique une personne.

4.2 Les attributs

Chaque entrée peut être considérée comme un objet (au sens C++) possédant donc certains attributs, par exemple si une personne est une entrée, les attributs peuvent être, le nom, le prénom, l'âge, ... On peut aussi définir des attributs obligatoires et d'autres optionnels, en d'autres termes, les attributs obligatoires devront être renseignés mais pas forcément les optionnels. Il existe par ailleurs pour chaque **DSE** des attributs d'administration qui ne servent qu'au serveur.

4.3 Les classes d'objet

On regroupe les objets qui sont du même domaine dans une classe d'object, celle-ci est caractérisée par des attributs obligatoires ou optionnels et un type. Les types de classe d'objet sont:
- type structurel car elle contient des d'objets concrets de l'annuaire (personnes, groupes de personnes, ...),
- type auxiliaire, c'est des classes d'objets qu'on peut créer, pour rajouter des informations (attributs) supplémentaires à des classes d'objet de type structurel déjà existantes. En C++ on dira que la classe auxiliaire dérive d'une classe structurelle,
-type abstraire, c'est les classes d'objet qui existent par défaut et qui n'ont pas de signification concrète, par exemple la classe top est la classe d'objet générique, toutes les autres classes dérivent de cette classe.

Le principe est donc le même qu'en C++, on retrouve une structure arborescente, avec à la racine la classe **top**, toutes les autres classes d'objet dérivent de cette classe générique, chaque classe hérite des propriétés d'une classe père et possède des attributs supplémentaires par rapport à ce dernier.

4.4 Les schémas

Un schéma décrit toutes les règles qu'utilisent le serveur **LDAP** pour décrire les classes d'objets (attributs, syntaxe, ...).

5 Installation d'OpenLDAP

Il existe de nombreux serveurs **LDAP**, nous utiliserons **OpenLDAP** qui comme son nom l'indique est sous licence GPL. J'ai choisi pour faire simple d'installer la version packagée de ma distribution, il faudra installer les packages **openldap-servers** et **openldap-clients**. Au besoin vous trouverez la dernière version stable à l'URL http://www.openldap.org.

6 Mettre en place son schéma d'annuaire

6.1 Mise en place des classes d'objet

Le fichier de conf **slapd.conf** fait appel à **/usr/local/etc/openldap/schema/core.shema** qui décrit les classes d'objet. Voilà un exemple avec la classe "**person**"

```
objectclass ( 2.5.6.6 NAME 'person'
        DESC 'RFC2256: a person'
        SUP top STRUCTURAL
        MUST ( sn $ cn )
        MAY ( userPassword $ telephoneNumber $ seeAlso $ description ) )
```

MUST correspond au attributs obligatoires et **MAY** à ceux facultatifs
objectClass est le nom de la classe qui descend elle même de la classe **top**
sn correspond à surname (nom)
cn correspond à common name (prénom nom)

Je vous laisse deviner la signification des autres attributs.

On voit qu'il est nécessaire de fournir les attributs **sn** (surname) et **cn** (common name), sont facultatifs le mot de passe (**userPassword**), le numéro de téléphone (**telephoneNumber**), les liens (**seeAlso**) et la description.

Les attributs sont définis dans le même fichier, la syntaxe est la suivante pour **telephoneNumber** par exemple :

```
attributetype ( 2.5.4.20 NAME 'telephoneNumber'
        DESC 'RFC2256: Telephone Number'
        EQUALITY telephoneNumberMatch
```

SUBSTR telephoneNumberSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.50{32})

Je vous présenterai la syntaxe plus tard, on peut dans un premier temps se limiter aux attributs disponibles. Pour créer une classe d'objet **breizhPerson** dérivant de **person**, disposant de l'attribut obligatoire **title** en plus et des arguments facultatifs **ou** (groupe de travail) et **l** (localisation). On tapera dans le fichier **core.shema** juste après la définition de la classe **person**

objectclass (2.5.6.6.2 NAME 'breizhPerson' SUP person STRUCTURAL
MUST (title)
MAY (ou $ l))

Vous noterez le nombre 2.5.6.6.2, ce nombre doit être unique dans le fichier, il dérive directement du numéro de la classe objet **person** qui a pour numéro 2.5.6.6. Il est évident que comme **breizhPerson** dérive de **person**, les attributs **sn** et **cn** sont aussi obligatoires.

A noter qu'avec une installation avec package les classes "locales" peuvent être créées dans le fichier **/etc/openldap/schema/local.schema**

6.2 Choix du suffixe

Le **rootDSE** ou suffixe correspond à l'entrée tout en haut de l'arbre (**DIT**) de l'annuaire, on utilise généralement le nom de domaine, avec la syntaxe suivante **dc=kervao, dc=fr** pour le domaine **kervao.fr** (**dc** correspond à Domain Component).

7 Configuration du serveur LDAP

On va créer un annuaire **LDAP** pour votre domaine privé **kervao.fr**. On doit modifier les fichiers **slapd.conf** et **ldap.conf** se trouvant sous **/etc/openldap**.Voilà pour le fichier de configuration **slapd.conf**

 # $OpenLDAP: pkg/ldap/servers/slapd/slapd.conf,v 1.8.8.6 2001/04/20 23:32:43 kurt Exp $
#
 # See slapd.conf(5) for details on configuration options.
 # This file should NOT be world readable.
#
 # Modified by Christian Zoffoli <czoffoli@linux-mandrake.com>
 # Version 0.2

include /usr/share/openldap/schema/core.schema
include /usr/share/openldap/schema/cosine.schema
include /usr/share/openldap/schema/corba.schema
include /usr/share/openldap/schema/inetorgperson.schema
include /usr/share/openldap/schema/java.schema
include /usr/share/openldap/schema/krb5-kdc.schema
include /usr/share/openldap/schema/kerberosobject.schema
include /usr/share/openldap/schema/misc.schema
include /usr/share/openldap/schema/nis.schema

```
include    /usr/share/openldap/schema/openldap.schema
include /usr/share/openldap/schema/autofs.schema
include /usr/share/openldap/schema/samba.schema
include /usr/share/openldap/schema/kolab.schema
include /usr/share/openldap/schema/evolutionperson.schema
include /usr/share/openldap/schema/calendar.schema
include /usr/share/openldap/schema/sudo.schema
include /usr/share/openldap/schema/dnszone.schema
include /usr/share/openldap/schema/dhcp.schema

include    /etc/openldap/schema/local.schema

# Define global ACLs to disable default read access and provide default
# behaviour for samba/pam use
include     /etc/openldap/slapd.access.conf

# Do not enable referrals until AFTER you have a working directory
# service AND an understanding of referrals.
#referral    ldap://root.openldap.org

pidfile       /run/ldap/slapd.pid
argsfile    /run/ldap/slapd.args

modulepath    /usr/lib64/openldap

# database backend modules available:
moduleload      back_mdb.la

# To allow TLS-enabled connections, create /etc/ssl/openldap/ldap.pem
# and uncomment the following lines.
#TLSRandFile           /dev/random
#TLSCipherSuite        HIGH:MEDIUM:+SSLv2
TLSCertificateFile      /etc/pki/tls/certs/ldap.pem
TLSCertificateKeyFile   /etc/pki/tls/private/ldap.pem
#TLSCACertificatePath   /etc/ssl/openldap/
#TLSCACertificateFile    /etc/ssl/cacert.pem
TLSCACertificateFile    /etc/pki/tls/certs/ldap.pem
#TLSVerifyClient never # ([never]|allow|try|demand)

# logging
#loglevel 256

#######################################################################
# database definitions
#######################################################################

database    mdb
suffix       "dc=kervao,dc=fr"
rootdn        "cn=Manager,dc=kervao,dc=fr"

# Cleartext passwords, especially for the rootdn, should
# be avoided.  See slappasswd(8) and slapd.conf(5) for details.
```

```
# Use of strong authentication encouraged.
rootpw    secret

# The database directory MUST exist prior to running slapd AND
# should only be accessable by the slapd/tools. Mode 700 recommended, répertoire où se
trouve la base de donnée
directory    /var/lib/ldap

# Tuning settings, please see the man page for slapd-bdb for more information
# as well as the DB_CONFIG file in the database directory
# commented entries are at their defaults
# In-memory cache size in entries
#cachesize 1000
# Checkpoint the bdb database after 256kb of writes or 5 minutes have passed
# since the last checkpoint
checkpoint 256 5

# Indices to maintain
index    objectClass                    eq

# persion-type searches
index    cn,mail,surname,givenname              eq,subinitial

# nss_ldap exact searches:
index    uidNumber,gidNumber,memberuid,member,uniqueMember    eq
# username completion via nss_ldap needs uid indexed sub:
index   uid                                  eq,subinitial

# samba:
index   sambaSID,sambaDomainName,displayName          eq

# autofs:
#index   nisMapName                            eq

# bind sdb_ldap:
#index   zoneName,relativeDomainName                  eq

# sudo
#index   sudoUser                              eq

# syncprov
#index  entryCSN,entryUUID                             eq


# Replicas running syncrepl as non-rootdn need unrestricted size/time limits:
limits group="cn=Replicator,ou=Group,dc=example,dc=com"
 size=unlimited
 time=unlimited

# Basic ACL (deprecated in favour of ACLs in /etc/openldap/slapd.access.conf)
access to attrs=userPassword
        by self write
        by anonymous auth
```

by dn="cn=Manager,dc=kervao,dc=fr" write
by * none

#

access to *
by dn="cn=Manager,dc=kervao,dc=fr" write
by * read

La base **LDAP** a été créée par défaut sous **/var/lib/ldap**

Le fichier **ldap.conf** peut être vide dans un premier temps voire inexistant.

Le mot de passe de l'administrateur est **secret** en clair, si ça ne vous convient pas et que vous voulez le crypter, il faudra taper

slappasswd

on saisit son mot de passe

New password:
Re-enter new password:

et voilà le résultat

{SSHA}vNCYRQthN6u3OqcTAJ4lt/9vIhjsFmI

A la place de

rootpw secret

Dans **slapd.conf**, vous mettrez donc:

rootpw {SSHA}vNCYRQthN6u3OqcTAJ4lt/9vIhjsFmI

8 Lancement du serveur

Pour le lancement avec **systemd** il existe un fichier **slapd.service** sous **/usr/lib/systemd/system/** voilà son contenu

After=syslog.target

[Service]
Type=forking
PIDFile=/run/ldap/slapd.pid
Environment="SLAPDURLLIST=ldap:/// ldapi:///" "LDAP_USER=ldap" "LDAP_GROUP=ldap" "SLAPDSYSLOGLOCALUSER=local4" "SLAPDSYSLOGLEVEL=0"
EnvironmentFile=/etc/sysconfig/slapd
ExecStartPre=/usr/share/openldap/scripts/ldap-config check
ExecStart=/usr/sbin/slapd -u ${LDAP_USER} -g ${LDAP_GROUP} -h ${SLAPDURLLIST} -l ${SLAPDSYSLOGLOCALUSER} -s ${SLAPDSYSLOGLEVEL}

[Install]
WantedBy=multi-user.target

maintenant pour que le service soit lancé à chaque boot de la machine il faudra taper

systemctl enable slapd.service

voilà le résultat

Created symlink from /etc/systemd/system/multi-user.target.wants/slapd.service to /usr/lib/systemd/system/slapd.service.

pour le lancer il suffit maintenant de taper

systemctl start slapd.service

et pour connaître son état

systemctl status slapd.service

voilà le résultat

● slapd.service - OpenLDAP Server Daemon
Loaded: loaded (/usr/lib/systemd/system/slapd.service; disabled; vendor preset: disabled)
Active: active (running) since Sat 2022-07-16 11:24:49 CEST; 2h 13min ago
Process: 3267 ExecStartPre=/usr/share/openldap/scripts/ldap-config check (code=exited, status=0/SUCCESS)
Process: 3302 ExecStart=/usr/sbin/slapd -u ${LDAP_USER} -g ${LDAP_GROUP} -h ${SLAPDURLLIST} -l ${SLAPDSYSLOGLOCALUSER} -s $ {SLAPDSYSLOGLEVEL} (code=exited, status=0/SUCCESS)
Main PID: 3303 (slapd)
Tasks: 3 (limit: 9283)
Memory: 3.1M
CPU: 57ms
CGroup: /system.slice/slapd.service
└─3303 /usr/sbin/slapd -u ldap -g ldap -h ldap:/// ldapi:/// -l local4 -s 0

juil. 16 11:24:48 serveur-slapd.kervao.fr systemd[1]: Starting OpenLDAP Server Daemon...
juil. 16 11:24:48 serveur-slapd.kervao.fr su[3274]: (to ldap) root on none
juil. 16 11:24:49 serveur-slapd.kervao.fr su[3274]: pam_unix(su:session): session opened for user ldap by (uid=0)
juil. 16 11:24:49 serveur-slapd.kervao.fr su[3274]: pam_unix(su:session): session closed for user ldap
juil. 16 11:24:49 serveur-slapd.kervao.fr ldap-config[3267]: Vérification de la configuration file /etc/openldap/slapd.conf : [OK]
juil. 16 11:24:49 serveur-slapd.kervao.fr systemd[1]: Started OpenLDAP Server Daemon.

8.1 Connexion chiffrée avec SSL

Pour une connexion chiffrée avec **SSL** entre le client et le serveur **LDAP** , tout se fait quasiment automatiquement avec une installation sous mageia avec le package **openldap-servers**. Cette installation va créer le certificat contenant la clé publique **/etc/pki/tls/certs/ldap.pem** et la clé privée **/etc/pki/tls/private/ldap.pem**. Le certificat est autosigné et n'est donc valable que pour un réseau privé. A noter que sur ma mageia 8, je retrouve les mêmes fichiers respectivement sous **/etc/ssl/certs** et **/etc/ssl/private**.

Dans le fichier **/etc/ldap/slapd.conf** on retrouvera les lignes suivantes qui pointent vers les bons fichiers

 # To allow TLS-enabled connections, create /etc/ssl/openldap/ldap.pem
 # and uncomment the following lines.
TLSCertificateFile /etc/pki/tls/certs/ldap.pem
TLSCertificateKeyFile /etc/pki/tls/private/ldap.pem
TLSCACertificateFile /etc/pki/tls/certs/ldap.pem

Il faudra également modifier le fichier **/etc/sysconfig/slapd** et rajouter la connexion **SSL** sur **LDAP**, **ldaps** comme ceci

 # SLAPD URL list
SLAPDURLLIST="ldap:/// ldapi:/// ldaps:///"

Si les fichiers n'ont pas été créés, il faudra installer le package **openssl-perl** pour créer un certificat perso auto signé qui ne marchera que sur un réseau privé comme j'ai pu le faire pour sendmail et dovecot. On va commencer par se créer un certificat CA.
On tapera pour cela

CA.pl -newca

voilà le résultat

CA certificate filename (or enter to create)

Making CA certificate ...
====
openssl req -new -keyout /etc/pki/CA/private/cakey.pem -out /etc/pki/CA/careq.pem
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/etc/pki/CA/private/cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:

j'ai mis un mot de passe à ce niveau

You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) [XX]:FR
State or Province Name (full name) []:Bretagne
Locality Name (eg, city) [Default City]:Brest
Organization Name (eg, company) [Default Company Ltd]:none
Organizational Unit Name (eg, section) []:none
Common Name (eg, your name or your server's hostname) []:serveur-slapd.kervao.fr
Email Address []:olivier.hoarau@funix.org

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:

j'ai mis un autre mot de passe ici, pensez à bien les noter ensuite !

An optional company name []:
==> 0
====
====
openssl ca -create_serial -out /etc/pki/CA/cacert.pem -days 1095 -batch -keyfile /etc/pki/CA/private/cakey.pem -selfsign -extensions v3_ca -infiles /etc/pki/CA/careq.pem
Using configuration from /etc/pki/tls/openssl.cnf
Enter pass phrase for /etc/pki/CA/private/cakey.pem:

on saisit le mot de passe PEM saisi plus haut

Check that the request matches the signature
Signature ok
Certificate Details:
Serial Number:
11:b4:74:e2:eb:82:f6:f5:08:7d:2d:2f:81:75:e3:1e:2e:70:08:e2
Validity
Not Before: Jul 16 11:48:56 2022 GMT
Not After : Jul 15 11:48:56 2025 GMT
Subject:
countryName = FR
stateOrProvinceName = Bretagne
organizationName = none
organizationalUnitName = none
commonName = serveur-slapd.kervao.fr
emailAddress = olivier.hoarau@funix.org
X509v3 extensions:
X509v3 Subject Key Identifier:
5F:EF:90:A6:C4:C5:17:65:77:EE:DA:62:14:23:A4:D8:98:5D:FF:C8
X509v3 Authority Key Identifier:
keyid:5F:EF:90:A6:C4:C5:17:65:77:EE:DA:62:14:23:A4:D8:98:5D:FF:C8

X509v3 Basic Constraints: critical
CA:TRUE
Certificate is to be certified until Jul 15 11:48:56 2025 GMT (1095 days)

Write out database with 1 new entries
Data Base Updated
==> 0
====
CA certificate is in /etc/pki/CA/cacert.pem

on copie maintenant les fichiers aux endroits qui vont bien (répertoire à créer éventuellement préalablement)

cp newcert.pem /etc/ssl/public/ldapcrt.pem

cp newreq.pem /etc/ssl/ldap/ldapkey.pem

chown ldap:ldap /etc/ssl/ldap/ldapkey.pem

l'utilisateur **ldap** doit en être propriétaire

chwon ldap:ldap /etc/ssl/ldap/ldapkey.pem

ça nous évitera une erreur du genre

slapd[5803]: main: TLS init def ctx failed: -1

maintenant on va modifier le fichier **/etc/openldap/slapd.conf** on rajoute les lignes suivantes

TLSCACertificateFile /etc/pki/CA/cacert.pem
TLSCertificateFile /etc/pki/CA/private/cakey.pem
TLSCertificateKeyFile /etc/pki/CA/cacert.pem

maintenant sur les postes clients on copiera le certificat signé du serveur LDAP où bon vous semble (dans mon exemple sous **/etc/pki/tls/certs/**). Il doit avoir les droits 644

chmod 644 ldapcrt.pem

On édite le fichier **/etc/openldap/ldapd.conf** et on modifie ainsi

 # SSL/TSL configuration. With CA-signed certs, TLS_REQCERT should be
 # "demand", with the CA certificate accessible
#TLS_REQCERT ([demand],never,allow,try)
 # We ship with allow by default as some LDAP clients (e.g. evolution) have
 # no interactive SSL configuration
TLS_REQCERT allow

 # CA Certificate locations
 # Use the default self-signed cert generated by openldap-server postinstall
 # by default
TLS_CACERT /etc/pki/tls/certs/ldap.pem

on relance **slapd**

systemctl stop sladp
systemctl start slapd

9 Utilisation sommaire

9.1 Ajouter un enregistrement

Vous avez différent moyen d'ajouter des données à l'annuaire, pour une meilleure compréhension on va d'abord aborder la méthode manuelle. Pour ajouter des données au serveur **LDAP** vous devez vous fournir un fichier au format **LDIF** (pour LDAP Directory

Interchange Format), le format est un format texte facilement lisible au contraire du format interne de l'annuaire. Voici un exemple de fichier **LDIF**, à noter que:
- chaque enregistrement dans le fichier est séparé du précédent et du suivant par une ligne vierge,
- les espaces sont pris en compte. **ATTENTION,** il est très important qu'il n'y ait aucun espace en fin de ligne. Dans ce cas vous risqueriez d'obtenir une erreur du style

ldap_add: Invalid syntax (21)
additional info: objectClass: value #0 invalid per syntax

La syntaxe du format **LDIF** est la suivante:

dn: description du distinguished name
objetclass: classe d'objet d'origine
...
objetclass: classe d'objet dérivée
type attribut: valeur
...

On va par exemple utiliser la classe **breizPerson** définie plus haut pour décrire une nouvelle personne **Veronique Hoarau** qu'on va rajouter dans l'annuaire. Elle appartient au service (**organizationalUnit**) **staff**, ce même service appartenant à l'organisation **kervao.fr**

Soit le fichier **entree.ldif**

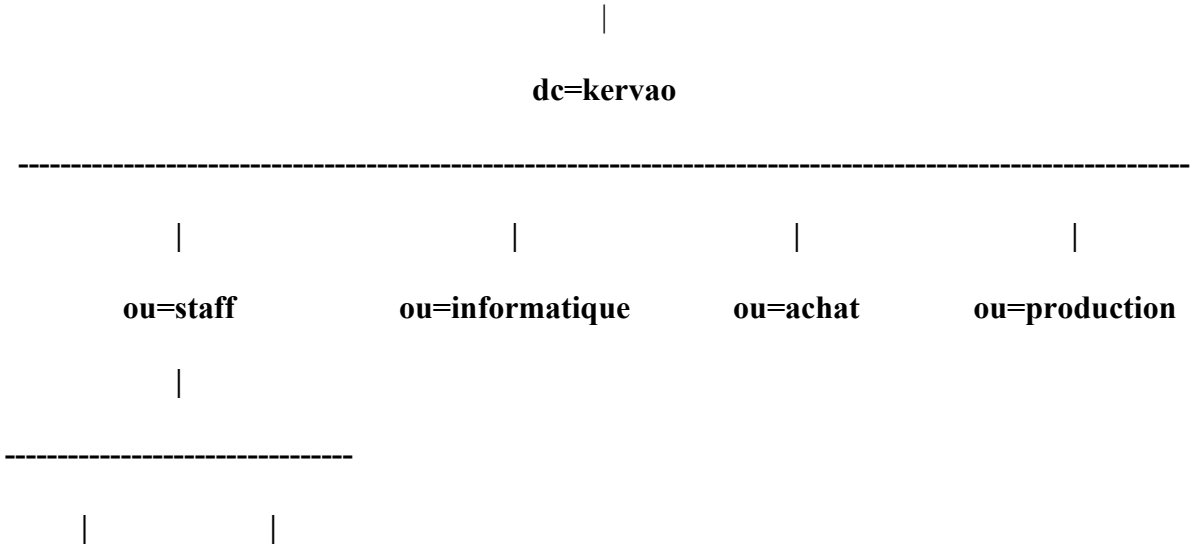
dn: dc=kervao, dc=fr
objectClass: dcObject
objectClass: organization
dc: kervao
o: kervao.fr

dn: ou=staff, dc=kervao, dc=fr
objectclass: organizationalUnit
ou: staff

dn: cn=Veronique Hoarau, ou=staff, dc=kervao, dc=fr
objectclass: person
objectclass: breizhPerson
cn: Veronique Hoarau
sn: Hoarau
title: madame

Quelques commentaires, le premier groupe correspond à la définition de votre organisation, le deuxième à celui du groupe de travail (**organizationalUnit**) et le dernier à la personne. Celle-ci est définie par son **dn** (Distinguished Name), on part du sommet **bz** (suffixe du nom de domaine), puis le nom de domaine, le groupe de travail et enfin la personne. L'arbre (**DIT**) pourrait ressembler à ça:

dc=fr



Au niveau de la définition de la personne:

objetclass: **person** définit la classe père de la classe **breizPerson**,
objetclass: **breizPerson** classe décrivant la personne,
cn et **sn** sont des attributs à renseigner obligatoirement,
title est un attribut obligatoire

On rajoutera l'enregistrement en utilisant la syntaxe suivante (en tant que simple utilisateur):

ldapadd -x -D "description du dn de l'administrateur" -W -f nom-du-fichier-ldif

Exemple concret:

ldapadd -x -D "cn=Manager, dc=kervao, dc=fr" -W -f entree.ldif
Enter LDAP Password: secret
adding new entry "dc=kervao, dc=fr"

adding new entry "ou=staff, dc=kervao, dc=fr"

adding new entry "cn=Veronique Hoarau, ou=staff, dc=kervao, dc=fr"

Pour rajouter par la suite un autre enregistrement dans le groupe **staff** , il sera plus nécessaire de rajouter la définition du groupe et de l'organisation. Soit le fichier **entree.ldif**

dn: cn=Olivier Hoarau, ou=staff, dc=kervao, dc=fr
objectclass: person
objectclass: breizhPerson
cn: Olivier Hoarau
sn: Hoarau
title: monsieur

On tape ensuite la commande:

ldapadd -x -D "cn=Manager, dc=kervao, dc=fr" -W -f entree.ldif
Enter LDAP Password:
adding new entry "cn=Olivier Hoarau, ou=staff, dc=kervao, dc=fr

9.2 Rechercher un enregistrement

On utilisera la fonction **ldapsearch**. Pour visualiser tout l'annuaire on peut taper :

ldapsearch -x -b 'dc=kervao, dc=fr' '(objectclass=*)'

Voilà le résultat

```
# extended LDIF
#
# LDAPv3
# filter: (objectclass=*)
# requesting: ALL
#

# kervao, fr
dn: dc=kervao, dc=fr
objectClass: dcObject
objectClass: organization
dc: kervao.fr
o: kervao.fr

# staff, kervao, fr
dn: ou=staff, dc=kervao, dc=fr
objectClass: organizationalUnit
ou: staff

# Veronique Hoarau, staff, kervao, fr
dn: cn=Veronique Hoarau, ou=staff, dc=kervao, dc=fr
objectClass: person
objectClass: breizhPerson
cn: Veronique Hoarau
sn: Hoarau
title: madame

# Olivier Hoarau, staff, kervao, fr
dn: cn=Olivier Hoarau, ou=staff, dc=kervao, dc=fr
objectClass: person
objectClass: breizhPerson
cn: Olivier Hoarau
sn: Hoarau
title: monsieur
```

 # search result
search: 2
result: 0 Success

 # numResponses: 5
 # numEntries: 4

9.3 Modifier un enregistrement

9.3.1 Rajouter un attribut à un enregistrement

On va rajouter l'attribut facultatif location (**l**) à l'enregistrement **Veronique Hoarau**. On va créer un fichier **modif.ldif** contenant:

dn: cn=Veronique Hoarau, ou=staff, dc=kervao, dc=fr
add: l
title: bureau36

On tape ensuite

ldapmodify -x -D "cn=Manager, dc=kervao, dc=fr" -W -f modif.ldif
Enter LDAP Password:secret
modifying entry "cn=Veronique Hoarau, ou=staff, dc=kervao, dc=fr"

9.3.2 Modifier un attribut

On va modifier l'attribut titre (**title**) à l'enregistrement **Veronique Hoarau**. On va créer un fichier **modif.ldif** contenant:

dn: cn=Veronique Hoarau, ou=staff, dc=kervao, dc=fr
changetype: modify
replace: title
title: mademoiselle

On tape ensuite

ldapmodify -x -D "cn=Manager, dc=kervao, dc=fr" -W -f modif.ldif
Enter LDAP Password:secret
modifying entry "cn=Veronique Hoarau, ou=staff, dc=kervao, dc=fr"

9.3.3 Supprimer un attribut

On va supprimer l'attribut location (**l**) à l'enregistrement **Veronique Hoarau**. On va créer un fichier **modif.ldif** contenant:

dn: cn=Veronique Hoarau, ou=staff, dc=kervao, dc=fr
delete: l

On tape ensuite

ldapmodify -x -D "cn=Manager, dc=kervao, dc=fr" -W -f modif.ldif
Enter LDAP Password:secret
modifying entry cn=Veronique Hoarau, ou=staff, dc=kervao, dc=fr

9.4 Supprimer un enregistrement

Pour supprimer l'enregistrement **Veronique hoarau**, on va créer un fichier **modif.ldif** contenant

dn: cn=Veronique Hoarau, ou=staff, dc=kervao, dc=fr
changetype: delete

On tape ensuite:

ldapmodify -x -D "cn=Manager, dc=kervao, dc=fr" -W -f modif.ldif
Enter LDAP Password:secret
deleting entry cn=Veronique Hoarau, ou=staff, dc=kervao, dc=fr

ATTENTION Vous ne pouvez pas supprimer un attribut obligatoire comme **title** pour la classe **breizhPerson**.

9.5 Exporter et importer une base de données LDAP

Le principe est d'exporter une base sur le serveur A pour la réimporter sur le serveur B. Pour exporter une base sur le serveur A on tapera

slapcat > base.ldif

Maintenant on revient sur le serveur B sur lequel on aura récupéré le fichier base.ldif, on stoppera tout d'abord slapd en tapant

systemctl stop slapd

puis pour importer

slapadd -l base.ldif

voilà le résultat

62d2baf8 /etc/openldap/slapd.conf: line 185: rootdn is always granted unlimited privileges.
62d2baf8 /etc/openldap/slapd.conf: line 189: rootdn is always granted unlimited privileges.
_#################### 100.00% eta none elapsed none fast!
Closing DB...

Attention si la base contient déjà des éléments préexistants en doublon, ça fera une erreur dans ce genre

62d2b3b7 /etc/openldap/slapd.conf: line 185: rootdn is always granted unlimited privileges.
62d2b3b7 /etc/openldap/slapd.conf: line 189: rootdn is always granted unlimited privileges.
 _# 8.38% eta 27s elapsed 02s spd 155.6 /s 62d2b3b9 mdb_id2entry_put: mdb_put failed: MDB_KEYEXIST: Key/data pair already exists(-30799) "dc=kervao,dc=fr"

62d2b3b9 => mdb_tool_entry_put: id2entry_add failed: err=-30799
62d2b3b9 => mdb_tool_entry_put: txn_aborted! MDB_KEYEXIST: Key/data pair already exists (-30799)
slapadd: could not add entry dn="dc=kervao,dc=fr" (line=1): txn_aborted! MDB_KEYEXIST: Key/data pair already exists (-30799)
.# 8.38% eta 27s elapsed 02s spd 0.0 /s
Closing DB..

Dans ce cas il faudra faire le ménage dans la base de donnée en la supprimant tout simplement

rm -f /var/lib/ldap/*

on relance **slapd** pour créer une base minimale

systemctl start slapd

puis on le recoupe pour taper

slapadd -l base.ldif

on relance ensuite **slapd**

systemctl start slapd

9.6 Passer en connexion chiffrée

Il suffit de rajouter l'option -Z pour passer en connexion chiffrée, avec le mode debug en plus (-d 1), on doit obtenir ce genre de message

TLS trace: SSL_connect:before SSL initialization
TLS trace: SSL_connect:SSLv3/TLS write client hello
TLS trace: SSL_connect:SSLv3/TLS write client hello
TLS trace: SSL_connect:SSLv3/TLS read server hello
TLS trace: SSL_connect:TLSv1.3 read encrypted extensions
TLS certificate verification: depth: 0, err: 0, subject: /CN=serveur-slapd.kervao.fr/OU=default ldap cert for serveur-slapd.kervao.fr/emailAddress=root@serveur-slapd.kervao.fr, issuer: /CN=serveur-slapd.kervao.fr/OU=default ldap cert for serveur-slapd.kervao.fr/emailAddress=root@serveur-slapd.kervao.fr
TLS trace: SSL_connect:SSLv3/TLS read server certificate
TLS trace: SSL_connect:TLSv1.3 read server certificate verify
TLS trace: SSL_connect:SSLv3/TLS read finished
TLS trace: SSL_connect:SSLv3/TLS write change cipher spec
TLS trace: SSL_connect:SSLv3/TLS write finished
ldap_sasl_bind
ldap_send_initial_request
ldap_send_server_request

(...)

TLS trace: SSL_connect:SSL negotiation finished successfully
TLS trace: SSL_connect:SSL negotiation finished successfully
TLS trace: SSL_connect:SSLv3/TLS read server session ticket
TLS trace: SSL_connect:SSL negotiation finished successfully
TLS trace: SSL_connect:SSL negotiation finished successfully
TLS trace: SSL_connect:SSLv3/TLS read server session ticket

(...)

ldap_free_connection 1 1
ldap_send_unbind
ber_flush2: 7 bytes to sd 3
TLS trace: SSL3 alert write:warning:close notify
ldap_free_connection: actually freed

il y a une erreur avec le certificat qui est auto signé mais comme on a mis **TLS_REQCERT allow** dans le fichier **ldap.conf** ce n'est pas bloquant**. C**ôté serveur cela donne cela

juil. 16 14:26:32 serveur-slapd.kervao.fr slapd[5289]: conn=1001 fd=11 ACCEPT from IP=127.0.0.1:37634 (IP=0.0.0.0:389)
juil. 16 14:26:32 serveur-slapd.kervao.fr slapd[5289]: conn=1001 op=0 EXT oid=1.3.6.1.4.1.1466.20037
juil. 16 14:26:32 serveur-slapd.kervao.fr slapd[5289]: conn=1001 op=0 STARTTLS
juil. 16 14:26:32 serveur-slapd.kervao.fr slapd[5289]: conn=1001 op=0 RESULT oid= err=0 text=
juil. 16 14:26:32 serveur-slapd.kervao.fr slapd[5289]: conn=1001 fd=11 TLS established tls_ssf=256 ssf=256
juil. 16 14:26:32 serveur-slapd.kervao.fr slapd[5289]: conn=1001 op=1 BIND dn="" method=128
juil. 16 14:26:32 serveur-slapd.kervao.fr slapd[5289]: conn=1001 op=1 RESULT tag=97 err=0 text=
juil. 16 14:26:32 serveur-slapd.kervao.fr slapd[5289]: conn=1001 op=2 SRCH base="dc=kervao,dc=fr" scope=2 deref=0 filter="(objectClass=*)"
juil. 16 14:26:32 serveur-slapd.kervao.fr slapd[5289]: conn=1001 op=2 SEARCH RESULT tag=101 err=0 nentries=4 text=
juil. 16 14:26:32 serveur-slapd.kervao.fr slapd[5289]: conn=1001 op=3 UNBIND
juil. 16 14:26:32 serveur-slapd.kervao.fr slapd[5289]: conn=1001 fd=11 closed

10Authentification des utilisateurs avec LDAP

10.1Présentation

L'authentification des utilisateurs sur le système se fait par défaut au moyen des fichiers **/etc/passwd** (définition des utilisateurs), **/etc/group** (identification des groupes d'utilisateurs) et éventuellement **/etc/shadow** si vous utilisez les "shadow password". C'est satisfaisant quand l'on dispose d'une machine isolée, par contre avec un parc d'une centaine de machines, il est peut concevable d'avoir à modifier ces fichiers sur tous les postes pour rajouter un utilisateur. L'idée est de centraliser l'authentification, NIS fait cela très bien ainsi que **LDAP,** c'est ce que l'on va voir dans ce paragraphe.

Dans la suite des opérations, on appellera serveur, la machine qui centralise la définition de tous les utilisateurs et groupes, le client fait appel au serveur pour l'authentification des utilisateurs.

10.2Installation

n récupérera sur ce site https://src.fedoraproject.org/repo/pkgs/openldap/MigrationTools-47.tar.gz le tarball **MigrationTools-4.7.tar.gz** on le décompresse en tapant

tar xvfz MigrationTools-4.7.tar.gz

Cela donne le répertoire **MigrationTools-47**

Maintenant on installera **nss-pam-ldapd** qui fournit un module Name Service Switch (**NSS**) qui accéder aux informations de compte et tous autres informations qu'on trouve dans les fichiers du système (**hosts**, **alias**, **netgroup**, etc...). Il fournit également un module **PAM** (Pluggable Authentification Module) pour un serveur **LDAP**.

Le daemon **nslcd** (Name Service LDAP Connection Daemon) fournit dans le package **nss-pam-ldapd** permet de gérer les connexions vers serveur **LDAP** via **PAM**. Pour le rendre actif à chaque démarrage on tapera donc

systemctl enable nslcd.service

voilà le résultat

Created symlink from /etc/systemd/system/multi-user.target.wants/nslcd.service to /usr/lib/systemd/system/nslcd.service.

on tachera de le lancer après avoir configurer le fichier **/etc/nslcd.conf** (voir plus bas).

10.3Configuration

10.3.1 Configuration d'un serveur

On modifiera si nécessaire le fichier **/usr/local/etc/openldap/slapd.conf** pour rajouter les règles d'accès d'usage:

```
# Basic ACL
access to attrs=userPassword
        by self write
        by anonymous auth
        by dn="cn=Manager,dc=kervao,dc=fr" write
        by * none
access to *
        by dn="cn=Manager,dc=kervao,dc=fr" write
        by * read
```

Relancer le serveur **LDAP**

systemctl restart slapd.service

A présent dans le répertoire **MigrationTools-47**, on va modifier le fichier **migrate_common.ph**, on doit y indiquer son nom de domaine, comme ceci :

```
# Default DNS domain
$DEFAULT_MAIL_DOMAIN = "kervao.fr";

# Default base
$DEFAULT_BASE = "dc=kervao,dc=fr";
```

Eventuellement vous pouvez modifier la ligne suivante spécifiant le serveur de mail bien que ce ne soit pas absolument nécessaire.

$DEFAULT_MAIL_HOST = "mail.padl.com";

A présent il faut rentrer les utilisateurs et groupes du système dans la base de données **LDAP**. Commençons d'abord par créer des fichiers temporaires au format **ldif.** On tape maintenant en tant que root (pour pouvoir lire **/etc/shadow**)

```
ETC_SHADOW=/etc/shadow
export ETC_SHADOW

./migrate_passwd.pl /etc/passwd /tmp/passwd.ldif
./migrate_group.pl /etc/group /tmp/group.ldif
```

Éditez les deux fichiers **ldif** pour ne laisser que les utilisateurs, enlever tous les utilisateurs et groupes système (root, lp, sys, apache, ...). Voici le contenu de mon **group.ldif** avec mon groupe utilisateur **hoarau**

```
dn: cn=hoarau,ou=Group,dc=kervao,dc=fr
objectClass: posixGroup
objectClass: top
cn: hoarau
userPassword: {crypt}*
gidNumber: 5000
memberUid: olivier
memberUid: veronique
```

Voici maintenant le contenu de mon **passwd.ldif** avec mes deux utilisateurs

```
dn: uid=olivier,ou=People,dc=kervao,dc=fr
uid: olivier
cn: olivier
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
userPassword: {crypt}$1$76UeLH8Z$K8rdYPRmUoiONZQm6hV4q.
shadowLastChange: 11858
shadowMax: 99999
shadowWarning: 7
shadowInactive: -1
shadowExpire: -1
shadowFlag: 1081428222
loginShell: /bin/bash
uidNumber: 5001
gidNumber: 5000
homeDirectory: /home/olivier
gecos: olivier

dn: uid=veronique,ou=People,dc=kervao,dc=fr
uid: veronique
cn: veronique
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
userPassword: {crypt}$1$OyedUoIU$uwpYR0bWJGzF4AFAHspSm/
shadowLastChange: 11858
shadowMax: 99999
shadowWarning: 7
```

shadowInactive: -1
shadowExpire: -1
shadowFlag: 1081428222
loginShell: /bin/bash
uidNumber: 5002
gidNumber: 5000
homeDirectory: /home/veronique
gecos: veronique

A présent il faudra créer le fichier **temp.ldif** qui va contenir la définition des Organizational Unit (**ou**) **Group** (groupe d'utilisateur) et **People** (utilisateur). Voici son contenu :

dn: dc=kervao,dc=fr
objectClass: dcObject
objectClass: organization
dc: kervao
o: kervao.fr

dn: ou=Group,dc=kervao,dc=fr
ou: Group
objectClass: top
objectClass: organizationalUnit
description: groupe d utilisateurs

dn: ou=People,dc=kervao,dc=fr
ou: People
objectClass: top
objectClass: organizationalUnit
description: utilisateurs du systeme

On peut commencer à rajouter tout cela (en mode verbeux), dans la base, on commence par les **ou Group** et **People**

ldapadd -v -x -D "cn=Manager,dc=kervao,dc=fr" -W -f /tmp/temp.ldif

voilà le résultat

ldap_initialize(<DEFAULT>)
Enter LDAP Password:
add objectClass:
dcObject
organization
add dc:
kervao
add o:
kervao.fr
adding new entry "dc=kervao,dc=fr"
modify complete

add ou:
Group
add objectClass:
top
organizationalUnit
add description:
groupe d utilisateurs
adding new entry "ou=Group,dc=kervao,dc=fr"

modify complete

add ou:
People
add objectClass:
top
organizationalUnit
add description:
utilisateurs du systeme
adding new entry "ou=People,dc=kervao,dc=fr"
modify complete

On continue avec le rajout des groupes et utilisateurs:

ldapadd -x -D "cn=Manager,dc=kervao,dc=fr" -f /tmp/passwd.ldif -W
Enter LDAP Password:
adding new entry "uid=olivier,ou=People,dc=kervao,dc=fr"

adding new entry "uid=veronique,ou=People,dc=kervao,dc=fr"

Puis

ldapadd -x -D "cn=Manager,dc=kervao,dc=fr" -f /tmp/group.ldif -W
Enter LDAP Password:
adding new entry "cn=hoarau,ou=Group,dc=kervao,dc=fr"

On visualise tout ça en tapant

ldapsearch -x -D "cn=Manager, dc=kervao, dc=fr" -W -b "dc=kervao,dc=fr"
Enter LDAP Password:
version: 2

#
 # filter: (objectclass=*)
 # requesting: ALL
#

 # kervao, fr
dn: dc=kervao, dc=fr
objectClass: dcObject
objectClass: organization
dc: kervao.fr
o: kervao.fr

 # Group, kervao, fr
dn: ou=Group,dc=kervao,dc=fr
ou: Group
objectClass: top
objectClass: organizationalUnit
description: groupe d utilisateurs

 # People, kervao, fr
dn: ou=People,dc=kervao,dc=fr
ou: People
objectClass: top
objectClass: organizationalUnit
description: utilisateurs du systeme

 # hoarau, Group, kervao, fr
dn: cn=hoarau,ou=Group,dc=kervao,dc=fr
objectClass: posixGroup
objectClass: top
cn: hoarau
gidNumber: 5000
memberUid: olivier
memberUid: veronique

 # olivier, People, kervao, fr
dn: uid=olivier,ou=People,dc=kervao,dc=fr
uid: olivier
cn: olivier
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
userPassword::
e2NyeXB0fSQxJDc2VWvMSDhaJEs4cmRVUFJtVW5pT05aUW02aFY0cS4=
shadowLastChange: 11858
shadowMax: 99999
shadowWarning: 7
shadowInactive: -1
shadowExpire: -1
shadowFlag: 1081428222
loginShell: /bin/bash
uidNumber: 5001
gidNumber: 5000
homeDirectory: /home/olivier
gecos: olivier

 # veronique, People, kervao, fr
dn: uid=veronique,ou=People,dc=kervao,dc=fr
uid: veronique
cn: veronique
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
userPassword::
e2NyeXB0fSQxJE95aWRVb0lVJHV3cFlOM7JXSkd6RjRBRkFIc3BTbS8=
shadowLastChange: 11858
shadowMax: 99999
shadowWarning: 7
shadowInactive: -1
shadowExpire: -1
shadowFlag: 1081428222
loginShell: /bin/bash
uidNumber: 5002
gidNumber: 5000
homeDirectory: /home/veronique
gecos: veronique

 # search result
search: 2
result: 0 Success

 # numResponses: 7
 # numEntries: 6

10.3.2 Configuration serveur et client

Pour un client sous mageia il faut installer les packages suivants

openldap
nss-pam-ldapd

Le client repose sur **nslcd** (Name Service LDAP Connection Daemon), à noter que sur un serveur si vous voulez également vous connecter sur le daemon LDAP, il faudra le configurer comme un client. On édite le fichier **/etc/nslcd.conf** et on apporte les modifications suivantes

 # The user and group nslcd should run as.
uid nslcd, ça n'a pas d'impact sur le fonctionnement
gid nslcd

 # le serveur LDAP
uri ldap://192.168.13.11/

 # The distinguished name of the search base qui doit être identique à la configuration de slapd
base dc=kervao,dc=fr

 # The distinguished name to bind to the server with qui doit être identique à la configuration de slapd
binddn cn=Manager,dc=kervao,dc=fr

 # The credentials to bind with, attention on doit mettre ici en clair le mot de passe qui a défini dans le fichier slapd.conf du serveur
bindpw mot-depasse-slapd-enclair

 # Customize certain database lookups.
base group ou=Group,dc=kervao,dc=fr
base passwd ou=People,dc=kervao,dc=fr
base shadow ou=People,dc=kervao,dc=fr

map shadow userPassword userPassword

Comme le mot de passe est en clair, il faudra veiller à ce que les droits du fichier soient bien à 600

-rw------- 1 root root 4855 juil. 17 08:04 /etc/nslcd.conf

Dans le fichier **/etc/nsswitch.conf** on modifiera les lignes suivantes pour lire

passwd: files nis ldap
shadow: files nis ldap
group: files nis ldap

A noter que le **nis** n'est pas nécessaire si vous n'avez pas mis en place de domaine NIS.

Bon maintenant on va éditer le fichier **/etc/pam.d/system-auth**. Pour information, c'est dans ce fichier qui va servir à appeler les bibliothèques qui vont bien pour l'authentification d'un utilisateur à la connexion sur une machine. Sans rentrer dans le détail, ce fichier et ceux qui se trouvent dans le même répertoire font parti du système **PAM** (Pluggable Authentication Modules), qui permet de gérer l'authentification sur le système que ce soit pour le login mais aussi pour d'autres services comme l'accès au serveur de mail, à **ftp**, etc.

Reprenons donc notre fichier **system-auth** il doit ressembler à ça (rajoutez uniquement les lignes concernant **pam_ldap** et laissez les autres) :

```
#%PAM-1.0

auth        required      pam_env.so
auth        sufficient    pam_tcb.so shadow nullok prefix=$2a$ count=8
auth        required      pam_deny.so
auth        sufficient    pam_ldap.so

account     sufficient    pam_tcb.so shadow
account     required      pam_deny.so
account     sufficient    pam_ldap.so

password    required      pam_cracklib.so try_first_pass retry=3 minlen=4 dcredit=0 ucredit=0
password    sufficient    pam_tcb.so use_authtok shadow write_to=shadow nullok prefix=$2a$ count=8
password    required      pam_deny.so
password    sufficient    pam_ldap.so

session     optional      pam_keyinit.so revoke
session     required      pam_limits.so
session     [success=1 default=ignore] pam_succeed_if.so service in crond quiet use_uid
-session    optional      pam_systemd.so
session     required      pam_tcb.so
session     sufficient    pam_ldap.so
```

Relancer **LDAP** sur le client (ou le serveur) maintenant

systemctl restart nslcd

ou lancer le s'il ne l'était pas à l'origine

systemctl start nslcd

A noter que le fichier **/etc/ldap.conf** ne sert plus à rien, vous pouvez l'ignorer superbement.

pour connaitre l'état du daemon on tapera

systemctl status nslcd.service

voilà le résultat

```
● nslcd.service - Naming services LDAP client daemon
     Loaded: loaded (/usr/lib/systemd/system/nslcd.service; enabled; vendor preset: disabled)
     Active: active (running) since Sun 2022-07-17 09:04:50 CEST; 5s ago
    Process: 84574 ExecStart=/usr/sbin/nslcd (code=exited, status=0/SUCCESS)
   Main PID: 84576 (nslcd)
      Tasks: 6 (limit: 2289)
```

Memory: 1.0M
CPU: 9ms
CGroup: /system.slice/nslcd.service
└─84576 /usr/sbin/nslcd

juil. 17 09:04:50 tahiti.kervao.fr systemd[1]: Starting Naming services LDAP client daemon...
juil. 17 09:04:50 tahiti.kervao.fr nslcd[84576]: version 0.9.11 starting
juil. 17 09:04:50 tahiti.kervao.fr nslcd[84576]: DEBUG: initgroups("nslcd",977) done
juil. 17 09:04:50 tahiti.kervao.fr nslcd[84576]: DEBUG: setgid(977) done
juil. 17 09:04:50 tahiti.kervao.fr nslcd[84576]: DEBUG: setuid(983) done
juil. 17 09:04:50 tahiti.kervao.fr nslcd[84576]: DEBUG: unlink() of /var/run/nslcd/socket failed (ignored): No such file or directory
juil. 17 09:04:50 tahiti.kervao.fr nslcd[84576]: accepting connections
juil. 17 09:04:50 tahiti.kervao.fr systemd[1]: Started Naming services LDAP client daemon.

si le serveur **LDAP** est configuré en connexion chiffrée, on modifiera ainsi le fichier **/etc/nslcd.conf**

 # on doit indiquer ici ldaps qui est le protocole LDAP sur SSL, j'ai dû indiquer le nom de mon serveur slapd car son certificat le référence par son nom et non par son adresse IP

uri ldaps://serveur-slapd.kervao.fr/

 # Use StartTLS without verifying the server certificate.
#ssl start_tls
#tls_reqcert allow

 # CA certificates for server certificate verification, on indique ici le chemin du certificat du serveur slapd qu'il faudra placer dans le répertoire /etc/openldap/cacerts du client
tls_cacertdir /etc/openldap/cacerts
tls_cacertfile /etc/openldap/cacerts/ldapcrt.pem

on stoppe **nslcd**, si on le relance en mode debug, voilà ce qu'on peut voir (extrait)

juil. 17 09:01:17 tahiti.kervao.fr nslcd[84462]: [43a858] <passwd=5008> DEBUG: ldap_initialize(ldaps://serveur-slapd.kervao.fr/)

c'est tout bon !

10.4Test de fonctionnement

C'est simple que ce soit sur le serveur ou le client, supprimer les lignes qui correspondent à vos utilisateur dans **/etc/passwd**, et **/etc/shadow** et faites de même pour vos groupes utilisateurs dans **/etc/group**. N'oubliez pas de faire une sauvegarde de ces fichiers au cas où ! Maintenant essayer de vous loguer en tant que simple utilisateur, et là normalement, vous devriez vous loguer sans problème.

10.5Gestion des utilisateurs et groupes

10.5.1 Créer un nouvel utilisateur

Maintenant pour créer un nouvel utilisateur, **useradd** ne fonctionne pas, car il repose uniquement sur **/etc/passwd**. On doit d'abord définir un mot de passe. On se sert pour cela de la fonction **slappasswd**

slappasswd -v -s toto345 -h {CRYPT}

On obtient

{CRYPT}rURm18fYhMvew

Il faudra créer un fichier **new.ldif** qui aura cette tête là :

dn: uid=utilisateur,ou=People,dc=kervao,dc=fr
uid: utilisateur
cn: utilisateur
objectClass: account
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
userPassword: {CRYPT}rURm18fYhMvew
shadowLastChange: 11858
shadowMax: 99999
shadowWarning: 7
shadowInactive: -1
shadowExpire: -1
shadowFlag: 1081428222
loginShell: /bin/bash
uidNumber: 5004
gidNumber: 5000
homeDirectory: /home/utilisateur
gecos: utilisateur

Pour mémoire voici la signification de chacun des paramètres des shadow passwords

shadowLastChange: date de dernière modification (en jour depuis le 1.1.70),
shadowMax: nombre de jours d'utilisation max du mot de passe (changement requis à l'issue), pas de période de validité si égal à 99999
shadowWarning: nombre de jours avant l'expiration pour avertir l'utilisateur ,
shadowInactive: nombre de jours après la date de l'expiration où on rend le compte inactif, fonctionnalité désactivé si égal à -1
shadowExpire: nombre de jours après le 1.1.70 où le compte sera désactivé, fonctionnalité désactivée si égal à -1
shadowFlag: ne sert à rien (dispo pour une utilisation future) .

ATTENTION Il ne doit pas y avoir de blanc ou de tabulation à la fin des lignes de votre fichier **ldif**

On tape **ldapadd** pour la saisie de l'utilisateur dans la base

ldapadd -x -D "cn=Manager,dc=kervao,dc=fr" -f /tmp/new.ldif -W
Enter LDAP Password:
adding new entry "uid=utilisateur,ou=People,dc=kervao,dc=fr"

C'est pas tout il faut indiquer maintenant que cet utilisateur appartient bien au groupe **hoarau** On crée ce fichier **groupe.ldif** contenant

dn: cn=hoarau,ou=Group,dc=kervao,dc=fr
Add: memberUid
memberUid: utilisateur

On modifie la base en tapant

ldapmodify -x -D "cn=Manager, dc=kervao, dc=fr" -W -f groupe.ldif
Enter LDAP Password:
modifying entry "cn=hoarau,ou=Group,dc=kervao,dc=fr"

Par contre il faudra créer manuellement la home directory en tant que root en tapant :

mkdir /home/utilisateur
chown -R utilisateur:hoarau /home/utilisateur
cp -R /etc/skel/* /home/utilisateur

10.5.2 Rajouter un groupe

Sans rentrer dans les détails des commandes, le fichier ldif à créer pour la saisie d'un nouveau groupe est le suivant

dn: cn=newgroupe,ou=Group,dc=kervao,dc=fr
objectClass: posixGroup
objectClass: top
cn: newgroupe
gidNumber: 5000

Voilà le fichier pour rajouter des utilisateurs au groupe

dn: cn=newgroupe,ou=Group,dc=kervao,dc=fr
Add: memberUid
memberUid: new-utilisateur

10.5.3 Supprimer un utilisateur

Pour supprimer l'utilisateur **utilisateur** on tapera

ldapdelete -x -D "cn=Manager, dc=kervao, dc=fr"
"uid=utilisateur,ou=People,dc=kervao,dc=fr" -W

10.6Changer son mot de passe

Maintenant le problème consiste à changer le mot de passe sans avoir à passer par le **Manager.** Un simple utilisateur doit pouvoir changer son propre mot de passe, prenons le cas de l'utilisateur **olivier,** il doit d'abord en trouver un avec la commande **slappasswd**

slappasswd -v -s tutu728 -h {CRYPT}

Voilà le résultat

{CRYPT}WW6h470hoW4nI

Il crée maintenant le fichier **modif.ldif** contenant

dn: uid=olivier, ou=People, dc=breizland, dc=bz
changetype: modify
replace: userPassword
userPassword: {CRYPT}WW6h470hoW4nI

Et il modifie la base en tapant

[olivier@asterix olivier]$ ldapmodify -x -D "uid=olivier, ou=People, dc=kervao, dc=fr" -f /tmp/modif.ldif -W
Enter LDAP Password: (mot de passe d'olivier)
modifying entry "uid=olivier, ou=People, dc=kervao, dc=fr"

Voilà son mot de passe a été changé. Vous me direz c'est bien compliqué, y a plus simple !

En tant que root modifiez le fichier **/etc/pam.d/passwd** (rajoutez uniquement les lignes concernant **pam_ldap** et laissez les autres) pour lire

#%PAM-1.0
auth include system-auth
auth sufficient pam_ldap.so
account sufficient pam_ldap.so
account include system-auth
password sufficient pam_ldap.so
password substack system-auth
password optional pam_gnome_keyring.so

Maintenant pour changer son mot de passe **olivier** tape le plus simplement du monde

passwd

Voilà le résultat

Enter login(LDAP) password:
New password:
Re-enter new password:
LDAP password information changed for olivier
passwd: all authentication tokens updated successfully

Ça marche !!!!

10.7Suggestion de mise en place

Pour l'administration je vous suggère **admin4** qu'on peut récupérer par ici http://www.admin4.org/ on le décompresse en tapant

tar xvfz admin4-3.0.1.tar.gz

cela donne le répertoire **admin4-3.0.1**. On installera préalablement les packages suivants

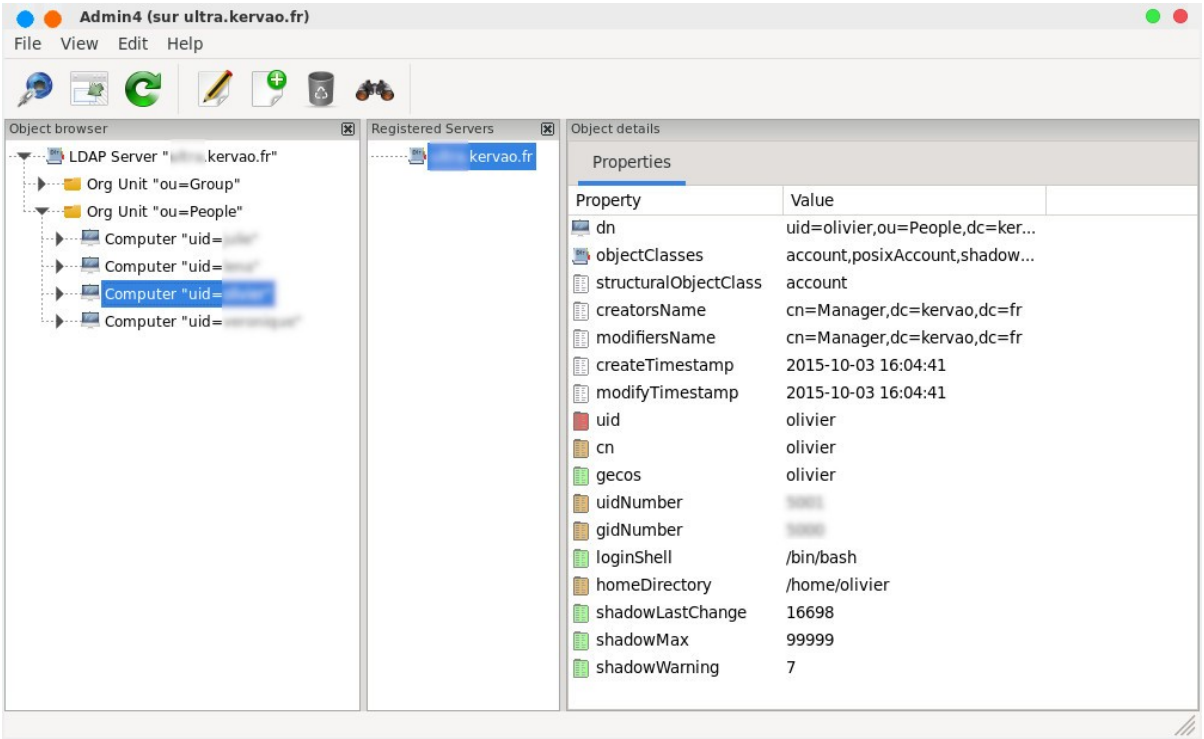
python-wxpython4
dnspython
python-requests
python-ldap

python3-psycopg2

on revient au répertoire **admin4-3.0.1** et on tape

python admin4.py

et voilà le résultat



pour être plus facilement accessible j'ai créé le script **/usr/bin/admin4** contenant

#!/bin/bash
python /usr/local/linux/systeme/admin4-3.0.1/admin4.py

en lui donnant les droits d'exécution

chmod 755 /usr/bin/admin4