Relazione esercitazione 2 Laboratorio di reti
Openldap

Franco Masotti Danny Lessio

March 23, 2015

Contents

Part I
Scelte di progetto

1 Ambiente di lavoro

`OpenLDAP` é presente nei repository di tutte principali distribuzioni `GNU/Linux`. Tuttavia abbiamo trovato particolarmente utile la documentazione presente sul wiki di `Arch Linux` in quanto esaustiva ma non dispersiva. Per questo motivo abbiamo provato `OpenLDAP` su sistemi derivati da questo. Si puó trovare la wiki per l'installazione su:

`http://wiki.archlinux.org/index.php/OpenLDAP`

e la gestione degli utenti e dei gruppi su:

`http://wiki.archlinux.org/index.php/LDAP_authentication`

2 Definizione del dominio e dell'utente amministratore

Per prima cosa abbiamo creato il nostro dominio che corrisponde alla radice dell'albero. Questo albero, chiamato anche `DIT` (`Directory Information Tree`), contiene tutte le entry dei gruppi e degli utenti, quindi é il nostro contenitore di informazioni. Ogni elemento dell'albero ha una chiave primaria che lo identifica. Questa viene chiamata `DN` (cioé `Distinguished Name`). Il `DN` per la radice é `gruppo2.labreti.it` che in notazione `LDIF` corrisponde a:

```
dn: dc=gruppo2,dc=labreti,dc=it
```

Abbiamo poi creato l'utente amministratore chiamato `root` che ha `DN` uguale a:

```
dn: cn=root,dn=gruppo2,dn=labreti,dn=it
```

Attraverso l'attributo `roleOccupant` stabiliamo il dominio su cui agisce, cioé `gruppo2.labreti.it`.

3 Gruppi e utenti

Successvamente abbiamo creato quattro `OU` (`organizationalUnit`) che rappresentano i quattro gruppi in cui suddividere le entry aventi caratteristiche in comune: `PEOPLE`, `GROUPS`, `HOST`, `DHCP`. L'`OU PEOPLE` contiene inizialmente le entry di quattro persone, mentre le altre `OU` serviranno successivamente.

Per generare le password degli utenti viene utilizzato il comando `slappasswd` che genera l'hash (con algoritmo `SSHA`, se non diversamente specificato) della password in input. Successivamente le password criptate vengono copiate nel file `users.ldif` e applicate come valore dell' attributo `userPassword`, presente in ogni utente. Se é giá presente un database, puó essere usata un'entry `add: userPassword` con `userPassword: <hashed user password>` nella riga successiva. Infine é necessario lanciare `ldapadd` oppure `ldapmodify` a seconda dei casi.

Abbiamo scelto la password ed il numero di telefono come dati sensibili. Per definire chi e come possa leggere questi dati, bisogna aggiungere delle direttive in `slapd.conf`. In questo modo per leggere i dati sensibili é necessario autenticarsi con gli switch `-W` e `-D`, altrimenti, con `-x` e `-b` tali dati vengono semplicemente omessi.

4 Script

Per automatizzare l'inizializzazione del database, il suo reset e per effettuare ricerche di prova abbiamo scritto alcuni script shell. Questo ci ha permesso di risparmiare parecchio tempo e di capire meglio il funzionamento del sistema `LDAP`.

5 Object classes e schemas

Per scegliere gli attributi che ci interessano, questi vanno scelti solo da alcune object class. Infatti esistono tre tipi di object class:

- `ABSTRACT`: classe `top` che identifica la fine di una gerarchia di classi. A questo tipo di classe, quindi, corrisponde solo la classe `top`.
- `AUXILIARY`: all'interno di ogni entry possiamo inserirne in numero arbitrario.
- `STRUCTURAL`: all'interno di ogni entry possiamo inserirne sono una che abbia come parent class (diretta) la classe `ABSTRACT` (cioé `top`), mentre possibile inserirne altre in cascata che abbiano come parent class una specifica classe. Ad esempio:
  - `inetOrgPerson` dipende da:

- `organizationalPerson` dipende da:
- `person` dipende da:
- `top`

Gli attributi e le object class sono definite negli schemi (cioé nei file `.schema` in `/etc/openldap/schema`). Ogni object class puó contenere piú attributi e ne definisce le proprietá, tra cui i vincoli `MUST` e `MAY`. Ogni attributo puó essere contenuto in piú object class.

6 LDAPS

Abbiamo creato uno script per la generazione del certificato con `openssl`. In questo modo tutte le operazioni di copia dei certificati e di configurazione del server sono automatizzate grazie a `make_cert.sh`. Il `DN` da immettere é quello deciso all'inizio, cioé `gruppo2.labreti.it`. Successivamente abbiamo usato due computer: uno con il server e l'altro con il client di `openldap`. Prima di effettuare la query abbiamo modificato il file `/etc/openldap/ldap.conf` in modo che il client possa accettare il nostro certificato server che non é firmato da `CAs` (`Certification Authorities`). Abbiamo poi verificato la connessione con TLS/SSL con la seguente query:

```
ldapsearch -x -b "dc=gruppo2,dc=labreti,dc=it" \
'(uid=*)'
```

Questo ritorna tutti gli utenti presenti nel database sul server all'indirizzo specificato nella direttiva `URI` in `/etc/openldap/ldap.conf`

Per eventuali problemi da parte del client con il certificato é sufficiente aggiungere una variabile d'ambiente subito prima di lanciare ldapsearch e/o specificare direttamente l'`URI` con lo switch `-H`, in questo modo:

```
LDAPTLS_REQCERT=allow ldapsearch -H ldaps://192.168.0.2:636 -x -b \
"dc=gruppo2,dc=labreti,dc=it" '(uid=*)'
```

É inoltre possibile aggiungere lo switch `-d1` per vedere lo scambio di chiavi.

Lanciando i comandi di ricerca con lo switch `-Z`, per abilitare il `TLS`, otteniamo un errore:

```
ldap_start_tls: Operations error (1)
additional info: TLS already started
```

Tuttavia nelle impostazioni del client abbiamo usato un `URI` con `ldaps` e con la porta `636`. In questo modo il servizio `TLS` é giá definito. Quindi lo switch `-Z` risulta ridondante perché tenta la duplice esecuzione di `TLS` e per questo il programma avvisa dell'errore. Lo switch, quindi, non va utilizzato.

Part II
Query di ricerca

É possibile effettuare le query di ricerca sia in locale sia in rete. Nel secondo caso é sufficiente aggiungere lo switch `-H` che identifica il protocollo (`LDAPS`) e l'host name o l'indirizzo ip del server. Ad esempio per effettuare una ricerca di un'utente via rete é sufficiente installare il pacchetto `openldap` su un secondo pc che fungerá da client. Successivamente si aggiunge una riga al per l'accettazione del certificato nel file di configurazione del client. Infine si effettua la ricerca vera e propria identificando la radice dell'albero `DIT` come base di ricerca:

```
sudo pacman -Sy openldap
sudo echo -e -n "TLS_REQCERT\tallow\n" >> /etc/openldap/ldap.conf
ldapsearch -H ldaps://192.168.0.2:636 -x '(uid=jacktripper)' -b \
"dc=gruppo2,dc=labreti,dc=it"
```

- `-H` identifica il protocollo, il server e la porta di ascolto remota.
- `-x` significa di utilizzare il metodo semplice di autenticazione (anonimo).
- `-b` identifica la base da cui incominciare la ricerca.

Possiamo specificare i search pattern in questo modo:

```
ldapsearch -x '(|(uid=jacktripper)(cn~=Danny))'
```

dove `|` é l'operatore `OR`. In questo modo vengono selezionati le entry aventi `uid=jacktripper` con le entry aventi `cn≅Danny` cioé `cn` simile a `Danny`. Gli operatori logici vengono quindi messi per primi rispetto rispetto alle coppie variabili valore.

Con la seguente query otteniamo tutti gli oggetti appartenenti all'`OU` people con quelli apparteneti all'object class `organizationalPerson`.

```
ldapsearch -x -b 'dc=gruppo2,dc=labreti,dc=it' \
'(&(ou=PEOPLE)(objectClass=organizationalPerson))'
```

Part III
Listati

7 slapd.conf

Questo file va posizionato in `/etc/openldap/slapd.conf`. Si tratta del file di configurazione del server. Le direttive aggiunte si trovano in coda al file.

```
#
# slapd.conf
#
# Copyright (C) 2016 frnmst (Franco Masotti) <franco.masotti@student.unife.it>
#                    dannylessio (Danny Lessio)
#
# This file is part of networks-lab.
#
# networks-lab is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# networks-lab is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with networks-lab.  If not, see <http://www.gnu.org/licenses/>.
#



#
# See slapd.conf(5) for details on configuration options.
# This file should NOT be world readable.
#
include /etc/openldap/schema/core.schema


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

pidfile /run/openldap/slapd.pid
```

```
argsfile /run/openldap/slapd.args

# Load dynamic backend modules:
# modulepath /usr/lib/openldap
# moduleload back_bdb.la
# moduleload back_hdb.la
# moduleload back_ldap.la

# Sample security restrictions
# Require integrity protection (prevent hijacking)
# Require 112-bit (3DES or better) encryption for updates
# Require 63-bit encryption for simple bind
# security ssf=1 update_ssf=112 simple_bind=64

# Root DSE: allow anyone to read it
# Subschema (sub)entry DSE: allow anyone to read it
# Other DSEs:
# Allow anonymous users to authenticate
# Directives needed to implement policy:
#
# allows anyone and everyone to read anything but restricts
#

#######################################################################
# BDB database definitions
#######################################################################

database bdb

# be avoid.  See slappasswd(8) and slapd.conf(5) for details.
# Use of strong authentication encouraged.
# The database directory MUST exist prior to running slapd AND
# Mode 700 recommended.
directory /var/lib/openldap/openldap-data

# Certificate/SSL Section
TLSCipherSuite  HIGH:MEDIUM:-SSLv2:-SSLv3
TLSCertificateFile  /etc/openldap/ssl/slapdcert.pem
TLSCertificateKeyFile  /etc/openldap/ssl/slapdkey.pem
suffix     "dc=gruppo2,dc=labreti,dc=it"
rootdn     "cn=root,dc=gruppo2,dc=labreti,dc=it"
rootpw {SSHA}ppdLDdVQG8leC3ZtCjtFKhFl33sVsGaA
include /etc/openldap/schema/cosine.schema
include /etc/openldap/schema/inetorgperson.schema
```

```
include /etc/openldap/schema/nis.schema
include /etc/openldap/schema/dnszone.schema
include /etc/openldap/schema/misc.schema
index   uid             pres,eq
index   mail            pres,sub,eq
index   cn              pres,sub,eq
index   sn              pres,sub,eq
index   dc              eq
index   relativeDomainName              eq
index   zoneName        eq
access to attrs=userPassword,telephoneNumber
by self write
by anonymous auth
by dn.base="cn=root,dc=gruppo2,dc=labreti,dc=it" write
by * none
access to *
by self write
by dn.base="cn=root,dc=gruppo2,dc=labreti,dc=it" write
by * read
```

8 ldap.conf

File di configurazione del client posizionato in `/etc/openldap/ldap.conf`.
Da notare la presenza di `TLS_REQCERT allow`

```
#
# ldap.conf
#
# Copyright (C) 2016 frnmst (Franco Masotti) <franco.masotti@student.unife.it>
#                    dannylessio (Danny Lessio)
#
# This file is part of networks-lab.
#
# networks-lab is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# networks-lab is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with networks-lab.  If not, see <http://www.gnu.org/licenses/>.
#


BASE dc=gruppo2,dc=labreti,dc=it
URI ldap://localhost
TLS_REQCERT allow
```

9 domain.ldif

Questo file é usato per la creazione del dominio e dei gruppi.

```
#
# domain.ldif
#
# Copyright (C) 2016 frnmst (Franco Masotti) <franco.masotti@student.unife.it>
#                    dannylessio (Danny Lessio)
#
# This file is part of networks-lab.
#
# networks-lab is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# networks-lab is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with networks-lab.  If not, see <http://www.gnu.org/licenses/>.
#



#
# Domain configuration file.
#

# gruppo2.labreti.it
dn: dc=gruppo2,dc=labreti,dc=it
dc: gruppo2
o: UNIFE
objectClass: dcObject
objectClass: organization

# root, gruppo2.labreti.it
dn: cn=root,dc=gruppo2,dc=labreti,dc=it
cn: root
description: LDAP administrator
objectClass: organizationalRole
objectClass: top
```

```
roleOccupant: dc=gruppo2,dc=labreti,dc=it

# PEOPLE, gruppo2.labreti.it
dn: ou=PEOPLE,dc=gruppo2,dc=labreti,dc=it
ou: PEOPLE
objectClass: organizationalUnit

# GROUPS, gruppo2.labreti.it
dn: ou=GROUPS,dc=gruppo2,dc=labreti,dc=it
ou: GROUPS
objectClass: organizationalUnit

# HOST, gruppo2.labreti.it
dn: ou=HOST,dc=gruppo2,dc=labreti,dc=it
ou: HOST
objectClass: organizationalUnit

# DHCP, gruppo2.labreti.it
dn: ou=DHCP,dc=gruppo2,dc=labreti,dc=it
ou: DHCP
objectClass: organizationalUnit
```

10 users.ldif

File usato per la creazione degli utenti.

```
#
# users.ldif
#
# Copyright (C) 2016 frnmst (Franco Masotti) <franco.masotti@student.unife.it>
#                    dannylessio (Danny Lessio)
#
# This file is part of networks-lab.
#
# networks-lab is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# networks-lab is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with networks-lab.  If not, see <http://www.gnu.org/licenses/>.
#



#
# Users configuration file.
#

dn: cn=Franco Masotti,ou=PEOPLE,dc=gruppo2,dc=labreti,dc=it
objectClass: top
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
ou: PEOPLE
mail: francomasotti@mail.gruppo2.labreti.it
telephoneNumber: 0532000001
uid: francomasotti
cn: Franco Masotti
sn: Masotti
userPassword: {SSHA}/XindgLhd8FcrGDH9JZUP6i2cSzcVkZH
```

```
uidNumber: 9000
gidNumber: 8000
homeDirectory: /home/francomasotti

dn: cn=Danny Lessio,ou=PEOPLE,dc=gruppo2,dc=labreti,dc=it
objectClass: top
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
ou: PEOPLE
mail: dannylessio@mail.gruppo2.labreti.it
telephoneNumber: 0532000002
uid: dannylessio
cn: Danny Lessio
sn: Lessio
userPassword: {SSHA}auDobwU09otHQ+c/4mG2BEbMviJmfuF2
uidNumber: 9001
gidNumber: 8000
homeDirectory: /home/dannylessio

dn: cn=Jack Tripper,ou=PEOPLE,dc=gruppo2,dc=labreti,dc=it
objectClass: top
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
ou: PEOPLE
mail: jacktripper@mail.gruppo2.labreti.it
telephoneNumber: 0532000003
uid: jacktripper
cn: Jack Tripper
sn: Tripper
userPassword: {SSHA}Es/R5EMCsHqPC3BUiUvn6VHC4vktlylq
uidNumber: 9002
gidNumber: 8000
homeDirectory: /home/jacktripper

dn: cn=Janet Wood,ou=PEOPLE,dc=gruppo2,dc=labreti,dc=it
objectClass: top
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
```

```
ou: PEOPLE
mail: janetwood@mail.gruppo2.labreti.it
telephoneNumber: 0532000004
uid: janetwood
cn: Janet Wood
sn: Wood
userPassword: {SSHA}hHe/I9pmZ+m8DfYEOhclcMUwMnlUyTAB
uidNumber: 9003
gidNumber: 8000
homeDirectory: /home/janetwood
```

11 search_examples.sh

Script per effettuare alcune query di ricerca.

```
#!/bin/bash

#
# search_examples.sh
#
# Copyright (C) 2016 frnmst (Franco Masotti) <franco.masotti@student.unife.it>
#                    dannylessio (Danny Lessio)
#
# This file is part of networks-lab.
#
# networks-lab is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# networks-lab is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with networks-lab.  If not, see <http://www.gnu.org/licenses/>.
#

#
# Search examples script.
#


echo -e -n "\n\n[INFO] Query examples started.\n\n"

# Test query.
echo -e -n "ldapsearch -x -b 'dc=gruppo2,dc=labreti,dc=it' '(objectClass=*)'\n"
ldapsearch -x -b 'dc=gruppo2,dc=labreti,dc=it' '(objectClass=*)'
echo -e -n "Press enter to continue.\n"
read

# Searching for a user (without password authentication).
echo -e -n "ldapsearch -x -b 'dc=gruppo2,dc=labreti,dc=it' '(cn=Franco \
Masotti)'\n"
```

```
ldapsearch -x -b 'dc=gruppo2,dc=labreti,dc=it' '(cn=Franco Masotti)'
echo -e -n "Press enter to continue.\n"
read

# Query with authentication to show sensible data.
echo -e -n "ldapsearch -W -D "cn=root,dc=gruppo2,dc=labreti,dc=it" \
'(uid=francomasotti)'\n"
ldapsearch -W -D "cn=root,dc=gruppo2,dc=labreti,dc=it" '(uid=francomasotti)'
echo -e -n "Press enter to continue.\n"
read

# Print db status.
echo -e -n "\n\n[INFO] Printing database status:\n\n"
slapcat

echo -e -n "\n\n[MAYBE OK] Query examples.\n\n"

# End script.
exit 0
```

12 commands.sh

Script che chiama gli altri script per la creazione del certificato, l'inizializzazione con i dati di esempio e le query di esempio.

```
#!/bin/bash

#
# commands.sh
#
# Copyright (C) 2016 frnmst (Franco Masotti) <franco.masotti@student.unife.it>
#                    dannylessio (Danny Lessio)
#
# This file is part of networks-lab.
#
# networks-lab is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# networks-lab is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with networks-lab.  If not, see <http://www.gnu.org/licenses/>.
#



#
# Server initialization script.
#

# Check if root is running this script.
if [ "$UID" -ne 0 ]; then
echo -e -n "You must be root tu run this script\n"
echo -e -n "sudo -u root $0\n"
exit 1
fi

#echo -en "Insert your root ldap password> "
#while [ "$ldapRootPwd" == "" ]; do
# read "ldapRootPwd"
```

```
#done
#echo -en "[DONE]\n"

./make_cert.sh
./add_example_entries.sh
#./search_examples.sh

# End script.
exit 0
```

13 make_cert.sh

Script per la creazione del certificato. L'amministratore si deve curare di fornire i dati corretti, sopratattutto il `DN`, altrimenti l'autenticazione fallisce.

```
#!/bin/bash

#
# make_cert.sh
#
# Copyright (C) 2016 frnmst (Franco Masotti) <franco.masotti@student.unife.it>
#                    dannylessio (Danny Lessio)
#
# This file is part of networks-lab.
#
# networks-lab is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# networks-lab is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with networks-lab.  If not, see <http://www.gnu.org/licenses/>.
#



#
# Create ssl certificate and enable it in the server.
#

# Define some variables.
sslCertPath="/etc/openldap/ssl"
slapdconf="/etc/openldap/slapd.conf"



# Check if root is running this script.
if [ "$UID" -ne 0 ]; then
echo -en "You must be root tu run this script\n"
echo -en "sudo -u root $0\n"
exit 1
```

```
fi

echo -en "SSL/TLS Openldap configuration started.\n"
systemctl stop slapd

echo -en "Creating SSL/TLS Openldap certificate... "


# Read certificate configuration.
OPENSSLCONFIG=${OPENSSLCONFIG- certificate.conf}

# Create cretificate whick lasts 3650 days.
openssl req -config $OPENSSLCONFIG -new -x509 -nodes -out slapdcert.pem \
-keyout slapdkey.pem -days 3650
#openssl req -new -x509 -nodes -out slapdcert.pem -keyout slapdkey.pem -days \
# 3650
echo -en "[DONE]\n"

echo -en "Removing previous certificate configuration..."
# Remove previous ssl and create it in openldap.
if [ -d "$sslCertPath" ]; then
rm -rf "$sslCertPath"
fi
mkdir -p "$sslCertPath"
echo -en "[DONE]\n"

echo -en "Moving certificate to $sslCertPath... "
# Move private and public key in ssl directory
mv slapdcert.pem slapdkey.pem "$sslCertPath/"

# Change permissions to files
chmod -R 755 "$sslCertPath/"
chmod 400 "$sslCertPath/slapdkey.pem"
chmod 444 "$sslCertPath/slapdcert.pem"
chown ldap "$sslCertPath/slapdkey.pem"
echo -en "[DONE]\n"

echo -en "Adding new certificate information... "
# Delete previous certificate info
sed -i "/Certificate\|TLS/ d" /etc/openldap/slapd.conf
# Write certificate info on slapd.conf
echo -e -n "# Certificate/SSL Section\nTLSCipherSuite \
HIGH:MEDIUM:-SSLv2:-SSLv3\nTLSCertificateFile \
$sslCertPath/slapdcert.pem\nTLSCertificateKeyFile \
```

```
$sslCertPath/slapdkey.pem\n" >> "$slapdconf"
echo -en "[DONE]\n"

echo -e -n "Adding new systemd configuration... "
if [ ! -f "/etc/systemd/system/slapd.service" ]; then
cp /usr/lib/systemd/system/slapd.service \
/etc/systemd/system/slapd.service
fi
# Delete previous configuration.
sed -i "/Service\|Type\|ExecStart/ d" /etc/systemd/system/slapd.service
# Write new config
echo -e -n "\n\
[Service]\n\
Type=forking\n\
ExecStart=/usr/bin/slapd -u ldap -g ldap -h \"ldap:/// ldaps:///\"\n" >> \
/etc/systemd/system/slapd.service

echo -en "[DONE]\n"

systemctl daemon-reload
systemctl start slapd

# Reset server configuration.
./restore_config.sh

echo -en "Remember to edit ldap.conf on the clients by adding:\nTSL_REQCERT \
\tallow\notherwise it will not be able to connect to this server.\n"
echo -en "[MAYBE OK] SSL/TLS Openldap configuration.\n"

exit 0
```

14 restore_config.sh

Script per il rispristino e la creazione della configurazione iniziale (reset) del database.

```
#!/bin/bash

#
# restore_config.conf
#
# Copyright (C) 2016 frnmst (Franco Masotti) <franco.masotti@student.unife.it>
#                    dannylessio (Danny Lessio)
#
# This file is part of networks-lab.
#
# networks-lab is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# networks-lab is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with networks-lab.  If not, see <http://www.gnu.org/licenses/>.
#



#
# Restore Openldap server configuration.
#

# Some variables
suffix="dc=gruppo2,dc=labreti,dc=it"
rootdn="cn=root,dc=gruppo2,dc=labreti,dc=it"
slapdconf="/etc/openldap/slapd.conf"
ldapconf="/etc/openldap/ldap.conf"
schemasDir="/etc/openldap/schema"
ret=""

ldapRootPwd="gruppo2"
```

```
# Check if root is running this script.
if [ "$UID" -ne 0 ]; then
echo -e -n "You must be root tu run this script.\n"
echo -e -n "sudo -u root $0\n"
exit 1
fi

echo -en "Restore configuration started...\n"

systemctl start slapd

# Wipe previous entries
# Only for Arch Linux and maybe other systems with systemd.
systemctl stop slapd
systemctl disable slapd

echo -en "Removing old database... "
# Remove database.
rm -rf /var/lib/openldap/openldap-data/*
echo -en "[DONE]\n"

echo -e -n "Adding new configuration in slapd.conf file... "
# Delete previous configuration.
# sed -i "... d" "$slapdcond"
sed -i "/suffix\|rootdn\|rootpw\|cosine.schema\|inetorgperson.schema\|nis.schema\|dn
# Write config to file.
echo "suffix     \"$suffix\"" >> "$slapdconf"
echo "rootdn     \"$rootdn\"" >> "$slapdconf"
echo -en "rootpw\t$(slappasswd -s "$ldapRootPwd")\n" >> "$slapdconf"
echo -en "\
include\t/etc/openldap/schema/cosine.schema\n\
include\t/etc/openldap/schema/inetorgperson.schema\n\
include\t/etc/openldap/schema/nis.schema\n\
include\t/etc/openldap/schema/dnszone.schema\n\
include\t/etc/openldap/schema/misc.schema\n" >> "$slapdconf"

# Delete previous configuration.
sed -i "/index/ d" "$slapdconf"
# Write config to file.
echo -e -n "\
index   uid             pres,eq\n\
index   mail            pres,sub,eq\n\
index   cn              pres,sub,eq\n\
```

```
index   sn              pres,sub,eq\n\
index   dc              eq\n\
index   relativeDomainName              eq\n\
index   zoneName        eq\n" >> "$slapdconf"

# Delete previous configuration.
sed -i "/access\|by/ d" "$slapdconf"
# Write config to file.
echo -e -n "\
access to attrs=userPassword,telephoneNumber\n\
\tby self write\n\
\tby anonymous auth\n\
\tby dn.base=\""$rootdn"\" write\n\
\tby * none\n\
access to *\n\
\tby self write\n\
\tby dn.base=\""$rootdn\"" write\n\
\tby * read\n\n" >> "$slapdconf"
echo -en "[DONE]\n"

# Copy user defined schema (to be used for DNS data).
cat "dnszone.schema" > "$schemasDir"/dnszone.schema

# Write to client config file.
echo -e -n "\
BASE "$suffix"\n\
URI ldap://localhost\n\
TLS_REQCERT allow\n" > "$ldapconf"

# Copy example db.
cp /etc/openldap/DB_CONFIG.example \
/var/lib/openldap/openldap-data/DB_CONFIG
chown ldap:ldap /var/lib/openldap/openldap-data/DB_CONFIG

rm -rf /etc/openldap/slapd.d/*

# Remove old config.
# Create db.
systemctl start slapd
systemctl stop slapd

echo -en "Checking slapd.conf file. Fatal errors will be \
reported... "
slaptest -f "$slapdconf" -F /etc/openldap/slapd.d/
```

```
echo -en "[DONE]\n"

chown -R ldap:ldap /etc/openldap/slapd.d
slapindex
chown ldap:ldap /var/lib/openldap/openldap-data/*

# Start ldap daemon.
systemctl enable slapd

echo -en "Starting slapd.service... "
systemctl start slapd
ret="$?"
if [ "$ret" -gt 0 ]; then
echo -en "[FAILED] Restore configuration.\n\n"
exit 1
fi

echo -en "[DONE]\n"

echo -en "[OK] Restore configuration."

exit 0
```

15 add_example_entries.sh

Script per l'aggiunta dei dati di esempio nel database `LDAP` sfruttando `domain.ldif` e `users.ldif`.

```
#!/bin/bash

#
# add_examples_entries.sh
#
# Copyright (C) 2016 frnmst (Franco Masotti) <franco.masotti@student.unife.it>
#                    dannylessio (Danny Lessio)
#
# This file is part of networks-lab.
#
# networks-lab is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# networks-lab is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with networks-lab.  If not, see <http://www.gnu.org/licenses/>.
#



#
# Server initialization script.
#

# Set some empty variables
ldapHost="127.0.0.1"
ret=""
ldapRootPwd="gruppo2"

# Check if root is running this script.
if [ "$UID" -ne 0 ]; then
echo -e -n "You must be root tu run this script\n"
echo -e -n "sudo -u root $0\n"
exit 1
```

```
fi

echo -en "Add example entries started..."

# Add groups and domain using domain.ldif as input file.
ldapadd -H ldaps://"$ldapHost":636 -x -w "$ldapRootPwd"  -D \
"cn=root,dc=gruppo2,dc=labreti,dc=it" -f \
domain.ldif
ret="$?"
#echo -e -n "Press enter to continue.\n"
#read

# Add users using users.ldif as input file.
ldapadd -H ldaps://"$ldapHost":636 -D "cn=root,dc=gruppo2,dc=labreti,dc=it" \
-w "$ldapRootPwd" -f users.ldif
ret="$(($ret+$?))"
#echo -e -n "Press enter to continue.\n"
#read

if [ "$ret" -gt 0 ]; then
echo -en "Add examples entries [FAILED]"
exit 1
fi

echo -en "[OK] Add example entries.\n"

# End script.
exit 0
```

16 slapd.service

Impostazioni di `systemd` per l'avvio di `slapd`. Da notare l'utilizzo esclusivo di `ldaps`.

```
#
# slapd.service
#
# Copyright (C) 2016 frnmst (Franco Masotti) <franco.masotti@student.unife.it>
#                    dannylessio (Danny Lessio)
#
# This file is part of networks-lab.
#
# networks-lab is free software: you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation, either version 3 of the License, or
# (at your option) any later version.
#
# networks-lab is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with networks-lab.  If not, see <http://www.gnu.org/licenses/>.
#


[Unit]
Description=OpenLDAP server daemon


[Install]
WantedBy=multi-user.target


[Service]
Type=forking
ExecStart=/usr/bin/slapd -u ldap -g ldap -h "ldap:/// ldaps:///"
```