OpenLDAP Cheat Sheet

Table of Contents

- Running *ldapadd/ldapmodify* with correct rootdn
- Running ldapsearch using simple authentication
- Running ldapsearch with SASL
- Specifying user’s password

Running *ldapadd/ldapmodify* with correct rootdn

Running `ldapadd` or `ldapmodify` and using the rootdn configured in `slapd.conf`:

```
$ ldapmodify -x -v -D 'cn=root,dc=fulltilt,dc=com' -w 'foo$bar'
$ ldapadd -x -v -D 'cn=root,dc=fulltilt,dc=com' -w 'foo$bar'
```

Assumes rootdn is defined something like this:

```
rootdn   "cn=root,dc=fulltilt,dc=com"
rootpw   {SSHA}u1zwxGiID0uDSA0p+jH+n7Ev5kHFMryq
```

where the encrypted password was created with slappasswd.

Running ldapsearch using simple authentication

```
$ ldapsearch -x -b 'dc=fulltilt,dc=com' 'userName=*'
```

Running `ldapsearch` using simple authentication and the rootdn. (Passwords won’t show up in the result unless bind is done this way.)

Prompt for password:

```
$ ldapsearch -D cn=root,dc=fulltilt,dc=com -W -x -b 'dc=fulltilt,dc=com' 'userName=*'
```

Specifying password on command line:

```
$ ldapsearch -D cn=root,dc=fulltilt,dc=com -w password -x -b 'dc=fulltilt,dc=com' 'userName=*'
```

Running ldapsearch with SASL

Make sure SASL stuff is in config. See sample `slapd.conf`, below. Then, run this command:

```
$ ldapsearch -v -U bclapper -b 'dc=fulltilt,dc=com'  username=*
```

Specifying user’s password

Easiest way is via LDIF, in a field. e.g.,

```
dn: cn=bmc,dc=fulltilt,dc=com
objectClass: localperson
cn: bmc
fullName: Brian Clapper
gn: Brian
sn: Clapper
mail: bmc@clapper.org
title: Chief Cook and Bottle Washer
userName: bmc
role: SuperUser
role: Owner
userPassword: bmc
```

`userPassword` field defines the password.

Must also configure `slapd` to look there. See “access to attr=userPassword” in sample config, below.

Sample `/etc/openldap/slapd.conf`:

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

# Local additions to the schema
include         /etc/openldap/schema/local.schema

# Allow LDAPv2 client connections.  This is NOT the default.
allow bind_v2

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

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

# Load dynamic backend modules:
# modulepath    /usr/sbin/openldap
# moduleload    back_bdb.la
# moduleload    back_ldap.la
# moduleload    back_ldbm.la
# moduleload    back_passwd.la
# moduleload    back_shell.la

# The next three lines allow use of TLS for encrypting connections using a
# dummy test certificate which you can generate by changing to
# /usr/share/ssl/certs, running "make slapd.pem", and fixing permissions on
# slapd.pem so that the ldap user or group can read it.  Your client software
# may balk at self-signed certificates, however.
# TLSCACertificateFile /usr/share/ssl/certs/ca-bundle.crt
# TLSCertificateFile /usr/share/ssl/certs/slapd.pem
# TLSCertificateKeyFile /usr/share/ssl/certs/slapd.pem

# 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

# Sample access control policy:
#       Root DSE: allow anyone to read it
#       Subschema (sub)entry DSE: allow anyone to read it
#       Other DSEs:
#               Allow self write access
#               Allow authenticated users read access
#               Allow anonymous users to authenticate
#       Directives needed to implement policy:
# access to dn.base="" by * read
# access to dn.base="cn=Subschema" by * read
# access to *
#       by self write
#       by users read
#       by anonymous auth
#
# if no access controls are present, the default policy
# allows anyone and everyone to read anything but restricts
# updates to rootdn.  (e.g., "access to * by * read")
#
# rootdn can always read and write EVERYTHING!

access to attr=userPassword
        by self write
        by anonymous auth
        by dn.base="cn=root,dc=fulltilt,dc=com" write
        by * none
access to *
        by self write
        by dn.base="cn=root,dc=fulltilt,dc=com" write
        by * read

#######################################################################
# ldbm and/or bdb database definitions
```

Generated by Jekyll.