Chapter 6: OpenLDAP using cn=config

cn=Config

Historically OpenLDAP has been statically configured, that is, to make a change to the configuration the slapd.conf file was modified and slapd stopped and started. In the case of larger users this could take a considerable period of time and had become increasingly unacceptable as an operational method. OpenLDAP version 2.3 introduced an optional new feature whereby configuration may be performed at run-time using a DIT entry called cn=config. The feature is known by a varied and confusing number of terms including **run-time configuration (RTC)** (our favorite), zero down-time configuration, cn=config and slapd.d configuration. First a number of points:

1. The feature is (at version 2.4) still optional which means that slapd.conf will continue to work.
2. OpenLDAP has a history of being quite brutal about withdrawing support of older capabilities. This means that migration to run-time configuration should be contemplated as soon as practical. Better to do it when you don't need the new release than to be forced to take a new release because of a critical bug AND have to migrate to the new configration regime.
3. Run-time configuration uses a configuration DIT (with a hardcoded suffix of cn=config) to control the operational configuration. Conceptually by modifying entries in this DIT (using an LDAP LDAP Browser or ldapadd'd LDIF files) immediate changes to slapd's operational behaviour are triggered without having to reload slapd as you would after making changes to slapd.conf.

Converting to use cn=Config

To create the cn=config DIT you can EITHER create a series of LDIF files that describe the configuration and apply them using slapadd OR convert your existing slapd.conf. If you enjoy pain we recommend the former approach. If you are a normal human being - convert slapd.conf using the process defined below. Conversion is a one time process (although it is reversable). From the time you have run the conversion the slapd.conf file is redundant. When loading slapd looks for the configuration directory (default slapd.d) and reads its configuration files from there and initializes the cn=config DIT. If the slapd.d directory is not found then slapd looks for slapd.conf.

To migrate/convert from slapd.conf to run-time configuration (RTC or cn=config) do the following:

1. Make sure the slapd.conf configuration file is stable and reflects the required functionality. This is a precautionalty measure and assumes familiarity with current slapd.conf based configuration. The last thing you want to do is to move to a major new feature and immediately require to use it. Best to have a bit of peace and tranquility and slowly acquire knowlege.

2. Stop the LDAP server.

3. Edit the slapd.conf file and add the following lines:

```
# before the first database definition
database config
# NOTE: the suffix is hardcoded as cn=config and
# MUST not have a suffix directive
# normal rules apply - rootdn can be anything you want
# but MUST be under cn=config
rootdn "cn=admin,cn=config"
# use any of the supported password formats e.g. {SSHA} etc
# or plaintext as shown
rootpw config
```

   There is a lot of drivel written about how to convert to slapd.d (cn=config) and it is all correct. However, if you want to actually use the cn=config feature (like reading and adding/changing attributes) using, say, an LDAP browser you must add the above lines.

4. Convert to run-time configuration (RTC) (cn=config) using the current slapd.conf file by running:

```
# uses slaptest (the most sensible method) but any utility which supports
# -f file and -F dir arguments will perform the conversion
# for example slapcat, slapadd etc.

# stop slapd
[fc]/etc/rc.d/init.d/slapd stop
# OR manually
killall slapd
# OR
[bsd]/usr/local/etc/rc.d/slapd.sh stop

[fc]cd /etc/openldap
[bsd]cd /usr/local/etc/openldap
# MUST - create standard default directory
mkdir slapd.d
# convert slapd.conf
slaptest -f slapd.conf -F slapd.d

# depending on the logged in user when you ran slaptest
# you may need to change ownership of slapd.d and all its files
chown -R ldap:ldap *

# rename slapd.conf
# this step is not necessary but is a useful
# precaution to ensure you access slapd.d
mv slapd.conf slapd.conf.bak

# start slapd
[fc]/etc/rc.d/init.d/slapd start
# OR manually
```

```
slapd -u ldap -g ldap

# if slapd fails to load use
slapd -d -1 -u ldap -g ldap
# and debug
```

Notes:

1. Unlike slapd.conf, cn=config files maintained in slapd.d require read and write permission for the user/group under which you run slapd (normally ldap). We ran the slaptest conversion under root which left us with unaccessible files and the load defaulted to use slapd.conf not cn=config. Assuming you are running slapd under user ldap (-u ldap) issue the chown for ldap:ldap as shown. File permissions of 0600 keep tight access to the files and should not require changing.

2. Renaming the slapd.conf file after running the slaptest conversion is not essential but does ensure that you are using cn=config and not defaulting to use slapd.conf due to some error. The downside of renaming is that your standard start scripts may assume slapd.conf certainly that was the case with our FreeBSD start script (/usr/local/etc/rc.d/slapd.sh) which failed to work because it assumes a slapd.conf file is present. We commented out the required_files line and the chown of slapd.conf line in the script - peace and tranquility reigned once more.

3. **Beware:** You can configure cn=config to an unusable state. We changed the rootdn of cn=config via an LDAP browser from cn=config to cn=admin (an invalid change since all config elements must end with a root of cn=config). The change was however accepted. The connection was immediately broken (correctly) but we could not bind under any value - old or new. We stopped and tried to start slapd which also failed because it refused to load under our newly modified rootdn (cn=admin). The only solution was to edit slapd.d/cn=config/olcDatabase={0}config.ldif and restore the oldRootDn attribute to cn=config. We then loaded slapd, changed the olcRootDn attribute via an LDAP browser to cn=admin,cn=config and everything worked perfectly.

   The rootDn of cn=config can be anything but it MUST be UNDER cn=config, for example cn=manager,cn=config will work, but cn=manager,cn=admin will be accepted but will fail to allow cn=config access and slapd will subsequently fail to load.

4. **Almost Conversion:** If you want to see what the future looks like but feel a tad nervous about conversion or just want to do some testing then edit the slapd.conf as shown in step 3 above (add the database config stuff). Do not run the conversion as shown in step 4 but just load slapd. You will able to log into cn=config and change attributes at run-time but when you stop and start slapd they will all be lost. Well we said it was almost a conversion.

5. If you are insatiably curious it is well worth looking through the files and directories in the slapd.d directory once you have done the conversion. One day you may need to repair 'em.

5. To access the cn=config feature from an LDAP browser assuming you have used the configuration above (otherwise modify the values used):

```
# Use the normal port and hostname settings with
# values shown are for LDAPBrowser/Editor
BaseDN cn=config
BindDN cn=admin,cn=config
password config

# alternatively you can use ldapsearch
ldapsearch -w config -x -D cn=admin,cn=config -b cn=config
```

6. If it all goes horribly wrong or you want to revert to old-fashioned(!) slapd.conf configuration (you have seen the future and it does not work for you) then the process is reversible:

```
# stop slapd
killall slapd
# OR
[fc]/etc/rc.d/init.d/slapd stop

[fc]cd /etc/openldap
[bsd]cd /usr/local/etc/openldap
# delete the slapd.d directory
rm -r slapd.d

# if you renamed slapd.conf as shown in the conversion
# procedure above you will need to restore slapd.conf
# then comment out the database config lines
# in slapd.conf
# now restart slapd
[fc]/etc/rc.d/init.d/slapd start
# OR manually
slapd -u ldap -g ldap
```

When you are feeling stonger and want to try again just repeat the conversion process. You can do this as often as you want. **Note:** If you modify the configuration in cn=config mode deleting the slapd.d directory (shown above) will lose all the changes. Depending on what changes you have made, say an index change under cn=config, then you may have some restoration work to do if you revert to slapd.conf. In general, limit your use or experimentation with cn=config features until you are committed to proceeding.

Using cn=config

The cn=config DIT can be read using standard LDAP command line tools such as ldapsearch and modified using ldapadd or ldapmodify which, IOHO, rather seems to defeat the objective of having a run-time configuration. The alternative is to use an LDAP browser to interactively read and write the attributes and entries. Access cn=config from LDAPBrowser/Editor.