Table of Contents

[Introduction 2](#_Toc463377967)

[Setting up OpenLDAP High Availability 2](#_Toc463377968)

[Pre-requisite 2](#_Toc463377969)

[HA OpenLDAP Sync Setup: 2](#_Toc463377970)

[Testing 5](#_Toc463377971)

[OpenLdap High Availability Setup with SSL/TLS 6](#_Toc463377972)

[Creation self-Signing Certificate 6](#_Toc463377973)

[Implementation Steps 6](#_Toc463377974)

[Update to openssl.conf 6](#_Toc463377975)

[Update unique serial number and index file for the keys to be generated 7](#_Toc463377976)

[Creating a Certifying Authority (CA) 7](#_Toc463377977)

[Creating a Client Certificate 8](#_Toc463377978)

[OpenLDAP Config Changes to add the certs 9](#_Toc463377979)

[Update Syncrepl with SecureLDAP changes 10](#_Toc463377980)

[How to Test 10](#_Toc463377981)

[Backup and Restore OpenLDAP with SSL and HA 11](#_Toc463377982)

[Backup 11](#_Toc463377983)

[Restore 11](#_Toc463377984)

[Restoring Users into the new LDAP server 14](#_Toc463377985)

# Introduction

This document address 3 main setup with OpenLDAP:

1. OpenLDAP High Availability.
2. Secure OpenLDAP (SSL) with HA.
3. Backup and Restore OpenLDAP with SSL and HA.

# Setting up OpenLDAP High Availability

## Pre-requisite

1. Make sure /etc/hosts has the correct ip address and the hostnames:
2. The master hostnames can be pinged from each other
3. Openldap client and servers are setup (yum install openldap-servers openldap-clients) and configured (addition of password to db files, db setup, bind dns setup etc…) and start ldap on the master nodes:

service slapd restart

1. Make sure the logs are captured in a valid log file for slapd

vi /etc/rsyslog.conf

Append this line:

# Point SLAPD logs to /var/log/slapd.log

local4.\* /var/log/slapd.log

Restart the rsyslogd service:

service rsyslog restart

1. Make sure to copy DB\_CONFIG as follows:

cp /usr/share/openldap-servers/DB\_CONFIG.example /var/lib/ldap/DB\_CONFIG

chown ldap:ldap /var/lib/ldap/DB\_CONFIG

## HA OpenLDAP Sync Setup:

The following ldif files needs to be created on the users home directory and should be executed on **all** the master nodes:

**NOTE**: Please change the cn , dc accordingly.

**vi basednupdate.ldif**

# updated your base dn below:

dn: olcDatabase={1}monitor,cn=config

changetype: modify

replace: olcAccess

olcAccess: {0}to \* by dn.base="gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth"

read by dn.base="cn=Manager,dc=srv,dc=world" read by \* none

**Execute:**

ldapadd -Y EXTERNAL -H ldapi:/// -f basednupdate.ldif

**Note:** If the above command fails, then edit the following file and update it and run

slaptest –u

**cd /etc/openldap/slapd.d/cn=config**

**vi olcDatabase={1}monitor.ldif**

**Change:**

olcAccess: {0}to \* by dn.base="gidNumber=0+uidNumber=0,cn=peercred,cn=extern

al,cn=auth" read by dn.base="**cn=grj,dc=ganeshrj,dc=com**" read by \* none

**vi syncproc\_module.ldif**

dn: cn=module,cn=config

objectClass: olcModuleList

cn: module

olcModulePath: /usr/lib64/openldap

olcModuleLoad: syncprov.la

**Execute:**

ldapadd -Y EXTERNAL -H ldapi:/// -f syncproc\_module.ldif

**vi syncproc.ldif**

dn: olcOverlay=syncprov,olcDatabase={2}hdb,cn=config

objectClass: olcOverlayConfig

objectClass: olcSyncProvConfig

olcOverlay: syncprov

olcSpSessionLog: 100

olcSpCheckpoint: 10 1

**Execute:**

ldapadd -Y EXTERNAL -H ldapi:/// -f syncproc.ldif

**vi addladpservers.ldif**

# Execute this on all master nodes. Make sure the server ids are changed for each node:

dn: cn=config

changetype: modify

replace: olcServerID

olcServerID: 1 ldap://ldap1.ganeshrj.com

olcServerID: 2 ldap://ldap2.ganeshrj.com

**Execute:**

ldapadd -Y EXTERNAL -H ldapi:/// -f addladpservers.ldif

**Here comes the major replication setup and need more attention:**

**vi master01.ldif**

dn: olcDatabase={2}hdb,cn=config

changetype: modify

add: olcSyncRepl

olcSyncRepl: rid=001 provider=ldap://ldap1.ganeshrj.com:389/ bindmethod=simple scope=sub binddn="cn=grj,dc=ganeshrj,dc=com" credentials=password searchbase="dc=ganeshrj,dc=com" schemachecking=off type=refreshAndPersist retry="5 10 30 +" interval=00:00:00:10

olcSyncRepl: rid=002 provider=ldap://ldap2.ganeshrj.com:389/ bindmethod=simple scope=sub binddn="cn=grj,dc=ganeshrj,dc=com" credentials=password searchbase="dc=ganeshrj,dc=com" schemachecking=off type=refreshAndPersist retry="5 10 30 +" interval=00:00:00:10

add: olcMirrorMode

olcMirrorMode: TRUE

add: olcDbIndex

olcDbIndex: entryCSN eq

olcDbIndex: entryUUID eq

**Execute:**

ldapadd -Y EXTERNAL -H ldapi:/// -f master01.ldif

**Note:**

**If the above command fails, then update it manually and run slaptest –u.**

cd /etc/openldap/slapd.d/cn=config/olcDatabase={2}hdb

vi olcOverlay={0}syncprov.ldif

olcSyncRepl: rid=001

             provider=ldap://ldap1.ganeshrj.com:389/

             bindmethod=simple

             scope=sub

             binddn="cn=grj,dc=ganeshrj,dc=com"

             credentials=password

             searchbase="dc=ganeshrj,dc=com"

             schemachecking=off

             type=refreshAndPersist

             retry="5 10 30 +"

             interval=00:00:00:10

olcSyncRepl: rid=002

             provider=ldap://ldap2.ganeshrj.com:389/

             bindmethod=simple

             scope=sub

             binddn="cn=grj,dc=ganeshrj,dc=com"

             credentials=password

             searchbase="dc=ganeshrj,dc=com"

             schemachecking=off

             type=refreshAndPersist

             retry="5 10 30 +"

             interval=00:00:00:10

olcDbIndex: entryUUID eq

olcDbIndex: entryCSN  eq

olcMirrorMode: TRUE

**Restart slapd service**

service slapd restart

Check the sladp.log file to see if both the servers are communicating.

## Testing

1. Do an LDAP search to see the users are getting pulled.

ldapsearch -x -b "dc=ganeshrj,dc=com"

1. Add users on one of the server and see if its replicated

vi Adduser.ldif

dn: uid=ganesh,ou=people,dc=testorg1,dc=ganeshrj,dc=com

objectclass:top

objectclass:person

objectclass:organizationalPerson

objectclass:inetOrgPerson

cn: Ganesh

sn: Ganesh

uid: ganesh

userPassword:ganesh-password

**Execute:**

ldapadd -x -D "cn=grj,dc=ganeshrj,dc=com" -W -f adduser.ldif

Then try ldapsearch on both the boxes to validate:

ldapsearch -x -b "dc=ganeshrj,dc=com"

**Load Balancing:**

Setup AWS LB with TCP ping/forward using port 389. Syntax to the LB provided below:

**ldapsearch -x -h internal-openldap-262926267.us-east-1.elb.amazonaws.com -p 389 -b "dc=ganeshrj,dc=com"**

# OpenLdap High Availability Setup with SSL/TLS

There are two parts to this.

1. Creation of Self Signing certificate for OpenLDAP.
2. Update OpenLDAP with Certs created and Update the Config and hdp databases with the certificate information.

## Creation self-Signing Certificate

### Implementation Steps

1. Create a CA First

2. Create a Client Openldap Cert

3. Sign the Client Openldap cert with the CA Created in Step 1

4. Use the cert (from step 3) with a ca cert (from Step 1) in open ldap

### Update to openssl.conf

This is “Nice to have” change since the default values are setup and don’t have to key in every time when a new key is generated.

Update */etc/pki/tls/openssl.conf* and change as follows (This is an example and can be tuned to organizational needs)

[ req\_distinguished\_name ]

countryName = Country Name (2 letter code)

#countryName\_default = XX

countryName\_default = US

countryName\_min = 2

countryName\_max = 2

stateOrProvinceName = State or Province Name (full name)

#stateOrProvinceName\_default = Default Province

stateOrProvinceName\_default = Virginia

localityName = Locality Name (eg, city)

#localityName\_default = Default City

localityName\_default = Ashburn

0.organizationName = Organization Name (eg, company)

#0.organizationName\_default = Default Company Ltd

0.organizationName\_default = Unknown Company Ltd

# we can do this but it is not needed normally :-)

#1.organizationName = Second Organization Name (eg, company)

#1.organizationName\_default = World Wide Web Pty Ltd

organizationalUnitName = Organizational Unit Name (eg, section)

#organizationalUnitName\_default =

Update date to 10 year validity

default\_days = 3650 # how long to certify for

Create a cert file on the need you need (Optional)

#certificate = $dir/cacert.pem # The CA certificate

certificate = $dir/ganeshrj.crt # The CA certificate

### Update unique serial number and index file for the keys to be generated

This step should be done only once.

cd /etc/pki/CA

touch serial

vi serial and add 01 for the 1st serial number

touch index.txt

### Creating a Certifying Authority (CA)

Now generate the key mentioned the openssl.conf (which is here: private\_key = $dir/private/cakey.key# The private key). This key will be used to generate a certificate.

openssl genrsa -des3 2048 > private/cakey.key

Using the key above create a CA certificate

openssl req -new -x509 -key private/cakey.key -out ganeshrj.crt -days 3560

### Creating a Client Certificate

Now that we have a certifying authority (CA), we need to create client certificate and have it signed by CA. A client can be an application or a browser etc… (OpenLDAP in our case now).

To generate a Client Key:

openssl genrsa -des3 2048 > openldap.key

Note: Make sure this key is not created with password, else the password needs to be keyed in every time when slapd is started. In order to create a key with no password , execute the following commands.

mv openldap.key openldap.key.pass

openssl rsa -in openldap.key.pass -out openldap.key

Now Create a Client Certificate using the key generated above:

openssl req -new -key openldap.key -out openldap.csr

Now Get the client Certificate signed by the CA created above.

**Note:-** The CA can be created once and the each master servers can generate their own certificate and get it signed by the CA.

Command in bold below:

[root@ldap1 ~]# **openssl ca -in openldap.csr -out openldap.crt**

Using configuration from /etc/pki/tls/openssl.cnf

Enter pass phrase for /etc/pki/CA/private/cakey.key:

Check that the request matches the signature

Signature ok

Certificate Details:

Serial Number: 1 (0x1)

Validity

Not Before: Aug 12 15:24:06 2016 GMT

Not After : Aug 10 15:24:06 2026 GMT

Subject:

countryName = US

stateOrProvinceName = Virginia

organizationName = Unknown Company Ltd

commonName = ldap1.ganeshrj.com

emailAddress = gr@gmail.com

X509v3 extensions:

X509v3 Basic Constraints:

CA:FALSE

Netscape Comment:

OpenSSL Generated Certificate

X509v3 Subject Key Identifier:

01:E9:C9:86:87:68:FE:5C:0B:3D:23:62:67:F2:01:59:78:F4:7E:43

X509v3 Authority Key Identifier:

keyid:9F:A2:BE:79:DE:52:4A:D1:C0:E6:64:79:70:32:99:41:75:E3:AE:24

Certificate is to be certified until Aug 10 15:24:06 2026 GMT (3650 days)

Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y

Write out database with 1 new entries

Data Base Updated

Now we have the client certificates available, we need to get the Certificates updated with OpenLDAP config files.

## OpenLDAP Config Changes to add the certs

Add the following certs to the database config file /etc/openldap/slapd.d/cn=config/olcDatabase={2}hdb.ldif

olcTLSCACertificateFile: /etc/pki/tls/certs/ganeshrj.crt

olcTLSCertificateFile: /etc/pki/tls/certs/openldap.crt

olcTLSCertificateKeyFile: /etc/pki/tls/certs/openldap.key

olcSecurity: tls=1

**Note**: You can also ldapmodify command, but the it might fail since the olcTLSCertificate attributes are not defined or unknown. So add these configurations to the DB files directly and run a slaptest –u to verify. You might end up with a checksum error, which is fine.

Update the slapd config files to use ldaps.

vi /etc/sysconfig/slapd

SLAPD\_URLS="ldapi:/// ldap:/// ldaps:///"

Update ldap.conf file with the certifying authority location.

/etc/openldap/ldap.conf

TLS\_CACERT /etc/pki/tls/certs/ganeshrj.crt

TLS\_REQCERT demand

**NOTE: VERY IMPORTANT:**

Make sure the CA certs, client CRT and client Key files are copied to the folder /etc/pki/tls/certs/, else the keys will not be recognized.

### Update Syncrepl with SecureLDAP changes

vi /etc/openldap/slapd.d/cn=config/olcDatabase={2}hdb/olcOverlay={0}syncprov.ldif

olcSyncRepl: rid=001 provider=ldaps://ldap1.ganeshrj.com:636/ bindmethod=simple scope=sub binddn="cn=grj,dc=ganeshrj,dc=com" credentials=password searchbase="dc=ganeshrj,dc=com" schemachecking=off type=refreshAndPersist retry="5 10 30 +" interval=00:00:00:10 tls\_reqcert=allow

olcSyncRepl: rid=002 provider=ldaps://ldap2.ganeshrj.com:636/ bindmethod=simple scope=sub binddn="cn=grj,dc=ganeshrj,dc=com" credentials=password searchbase="dc=ganeshrj,dc=com" schemachecking=off type=refreshAndPersist retry="5 10 30 +" interval=00:00:00:10 tls\_reqcert=allow

Now restart slapd to get the SSL/TLS into effect.

service slapd restart

Make sure the same steps are executed on the other master server too.

### How to Test

Try executing ldapsearch (with out –ZZ), you should get an TLS validation error. The ldapsearch should be executed with –ZZ option to query against a secured LDAP.

ldapsearch -ZZ -x -b "dc=ganeshrj,dc=com"

# Backup and Restore OpenLDAP with SSL and HA

## Backup

To Backup the existing OpenLDAP, run the following commands:

## To Backup the Configurations and databases.

sudo slapcat –n 0 > /tmp/config.ldif

## To Backup the existing users and groups.

sudo slapcat –n 2 > /tmp/users.ldif

## Restore

Please note that the config.ldap has the syncrepl and TLSCertificates attributes, manual intervention is needed to remove those attributes and update them back.

1. Stop the LDAP server (service slapd stop) and follow prerequisite points 1 to 5 (if not implemented).
2. Make sure the CA Certs and signed certs that are referenced in the config.ldif are present in /etc/pki/tls/certs/ directory.
3. Now, update the config.ldif, search and point the ldap to new server FDQN (if the hostname changed) and make the following changes.

**NOTE:-**

While restoring, OpenLDAP will not honor, the olcTLSCACertificate and olcSyncrepl attributes.  This  looks like a known issue on the OpenLDAP side. So remove the same and add it manually once the initial pull is completed.

Lines to remove from config.ldif file

olcTLSCACertificateFile: /etc/pki/tls/certs/ca.crt

olcTLSCertificateFile: /etc/pki/tls/certs/openldap.crt

olcTLSCertificateKeyFile: /etc/pki/tls/certs/openldap.key

olcSecurity: tls=1

AND

olcSyncrepl: rid=001 provider=ldaps://ldap1.ganeshrj.com:636/ bindmethod=simple scope=sub binddn="cn=grj,dc=ganeshrj,dc=com" credentials=password searchbase="dc=ganeshrj,dc=com" schemachecking=off type=refreshAndPersist retry="5 10 30 +" interval=00:00:00:10 tls\_reqcert=allow

olcSyncrepl: rid=002 provider=ldaps://ldap2.ganeshrj.com:636/ bindmethod=simple scope=sub binddn="cn=grj,dc=ganeshrj,dc=com" credentials=password searchbase="dc=ganeshrj,dc=com" schemachecking=off type=refreshAndPersist retry="5 10 30 +" interval=00:00:00:10 tls\_reqcert=allow

olcMirrorMode: TRUE

olcDbIndex: entryCSN eq

olcDbIndex: entryUUID eq

1. Update the correct olcRootPW generated with slappasswd command.
2. Run a slaptest –u to make sure  no other errors occur except checksum.
3. Make sure there are no files or directories under /etc/openldap/slapd.d  folder.  If exists, then delete the same.
4. Execute the following command now on the directory where we have the updated config.ldif … This should create a bunch of config files and DBs from the existing config.ldif..

slapadd -F /etc/openldap/slapd.d -b cn=config -l config.ldif

You should see 100% DB import as follows:

\_#################### 100.00% eta   none elapsed            none fast!

Closing DB...

1. Put back the changes to the below files:

/etc/openldap/slapd.d/cn=config/olcDatabase={2}hdb.ldif

Add back the lines: NOTE:- Make sure you have the new keys generated for the servers to be restored and provide the new keys here.

olcTLSCACertificateFile: /etc/pki/tls/certs/ca.crt

olcTLSCertificateFile: /etc/pki/tls/certs/openldap.crt

olcTLSCertificateKeyFile: /etc/pki/tls/certs/openldap.key

olcSecurity: tls=1

and:

/etc/openldap/slapd.d/cn=config/olcDatabase={2}hdb/olcOverlay\=\{0\}syncprov.ldif

olcSyncrepl: rid=001 provider=ldaps://ldap1.ganeshrj.com:636/ bindmethod=simple scope=sub binddn="cn=grj,dc=ganeshrj,dc=com" credentials=password searchbase="dc=ganeshrj,dc=com" schemachecking=off type=refreshAndPersist retry="5 10 30 +" interval=00:00:00:10 tls\_reqcert=allow

olcSyncrepl: rid=002 provider=ldaps://ldap2.ganeshrj.com:636/ bindmethod=simple scope=sub binddn="cn=grj,dc=ganeshrj,dc=com" credentials=password searchbase="dc=ganeshrj,dc=com" schemachecking=off type=refreshAndPersist retry="5 10 30 +" interval=00:00:00:10 tls\_reqcert=allow

olcMirrorMode: TRUE

olcDbIndex: entryCSN eq

olcDbIndex: entryUUID eq

1. Now, update the /etc/openldap/ldap.conf file to add the keys:

TLS\_CACERT      /etc/pki/tls/certs/ca.crt  = You CA CERT here

TLS\_REQCERT     demand

1. Update the /etc/sysconfig/slapd to include ladps:/// :

#SLAPD\_URLS="ldapi:/// ldap:///"

SLAPD\_URLS="ldapi:/// ldap:/// ldaps:///"

1. Change ownership of slapd.d directory back to ldap since it will be created with root user.

chown -R ldap:ldap /etc/openldap/slapd.d/

1. Start the slapd service and check the logs to see the TLS is up. Once the setup is done on the other master, then you should see the replication messages on the log file.
2. Repeat the steps on server 2.

## Restoring Users into the new LDAP server

1. Stop the LDAP server.
2. Run the following command to import the users.ldif

slapadd –n 2 –l users.ldif

1. Make sure the newly added User databases have the right ownership.

cd /var/lib/ldap

chown ldap:ldap \*

1. Start the LDAP server.
2. Run the ldapsearch command to see if the users are retrieved.
3. Switch over to the second master server and run the same LDAP search and you should see the users restored.