

Install Kopano Core on Debian or Ubuntu with OpenLDAP

- Introduction
 - Tested
- Installation
 - Install mysql and apache / php
 - Install OpenLDAP
 - Create the placeholder for our users.
 - Download and install kopano
 - Add an kopano user to our ldap
- Ldap optimization
 - Backup
 - Restore
- Disable anonymous binding
 - Changing the default behaviour
 - Testing if it works

Introduction

A quick guide to get you up to speed regarding kopano with the (open)LDAP backend.

Tested

This guide has been tested on Debian 7, 8 and Ubuntu 14.04 LTS,

Installation

Please note : slapd uses the hostname to "guess" the organisation name for the LDAP.

In this howto we use the hostname kopano.example.local, which results in dc=example,dc=local in LDAP.

For your setup you probably want to change this to something more useful.

Install mysql and apache / php

```
$ sudo apt-get install mysql-server libapache2-mod-php5
```

Enter the mysql password twice and write it down.

Install OpenLDAP

```
$ sudo apt-get install slapd ldap-utils
```

Enter LDAP administrator password twice and write it down.

Use slapcat to verify that the install took the defaults from your hostname for your organisation correctly, if that's not the case run **dpkg-reconfigure -p low slapd** and set it up manually

```
$ sudo slapcat
dn: dc=example,dc=local
objectClass: top
objectClass: dcObject
objectClass: organization
o: example.local
dc: example
structuralObjectClass: organization
entryUUID: 907f25dc-91f2-1032-97fa-b34646bf14f6
creatorsName: cn=admin,dc=example,dc=local
createTimestamp: 20130805081250Z
entryCSN: 20130805081250.289774Z#000000#000#000000
modifiersName: cn=admin,dc=example,dc=local
modifyTimestamp: 20130805081250Z

dn: cn=admin,dc=example,dc=local
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator
userPassword:: e1NTSEF9dm1rV2lGdmVPbVBXTnI4blhSbE5oeVVmTTVSWm4vV2U=
structuralObjectClass: organizationalRole
entryUUID: 907fc91a-91f2-1032-97fb-b34646bf14f6
creatorsName: cn=admin,dc=example,dc=local
createTimestamp: 20130805081250Z
entryCSN: 20130805081250.293957Z#000000#000#000000
modifiersName: cn=admin,dc=example,dc=local
modifyTimestamp: 20130805081250Z
```

Create the placeholder for our users.

Create a file called org.ldif containing:

```
dn: ou=People,dc=example,dc=local
objectClass: organizationalUnit
objectClass: top
ou: People
```

Import the ldif file into ldap.

```
$ sudo ldapadd -x -D cn=admin,dc=example,dc=local -W -f org.ldif
```

Check if it was added with a simple search.

```
$ sudo ldapsearch -x -D cn=admin,dc=example,dc=local -W -b
dc=example,dc=local
```

Download and install kopano

Choose the version for your distribution from <https://download.kopano.io/supported/core:/final/>

We will be using https://download.kopano.io/supported/core:/final/Debian_8.0/

Note: The user and password are those of your Kopano portal account.

```
$ sudo echo 'deb
https://serial:<ENTERYOURSERIALHERE>@download.kopano.io/supported/core:/fi
nal/Debian_8.0/ .' > /etc/apt/sources.list.d/kopano.list

$ sudo curl
https://serial:<ENTERYOURSERIALHERE>@download.kopano.io/supported/core:/fi
nal/Debian_8.0/Release.key | apt-key add -

$ sudo apt update
$ sudo apt install kopano-server-packages
```

After the boot check if kopano is up and running.

```
$ sudo kopano-admin -l

User list for Default(1):
Username Fullname Homeserver
-----
SYSTEM SYSTEM kopano
```

Add the kopano schema to our ldap

```
$ sudo zcat /usr/share/doc/kopano/kopano.ldif.gz | ldapadd -H ldapi:/// -Y
EXTERNAL
```

Add an kopano user to our ldap

Create a new Idif file called user.ldif containing the following. This user will have kopano admin rights:

```
dn: uid=john,ou=People,dc=example,dc=local
objectClass: posixAccount
objectClass: top
objectClass: kopano-user
objectClass: inetOrgPerson
gidNumber: 1000
cn: John Doe
homeDirectory: /home/john
mail: john@example.local
uidNumber: 1000
kopanoAliases: j.doe@example.local
kopanoUserServer: kopano
uid: john
kopanoAccount: 1
kopanoAdmin: 1
sn: Doe
userPassword: john
kopanoQuotaOverride: 1
kopanoEnabledFeatures: imap
kopanoDisabledFeatures: pop3
kopanoQuotaWarn: 1000000000
kopanoQuotaSoft: 1100000000
kopanoQuotaHard: 1200000000
```

Verify the user anonymously.

```
$ sudo ldapsearch -xLLL -b dc=example,dc=local uid=john
```

Changing the kopano configuration

Edit /etc/kopano/server.cfg

Change the line user_plugin into the following.

```
user_plugin          = ldap
```

|

Setup the ldap.cfg

Depending on the Kopano version (8.2.0 and higher) :

```
$ sudo cd /etc/kopano/
$ sudo cp /usr/share/doc/kopano/example-config/ldap.cfg ldap.cfg
```

And include the correct ldap template and change the other required fields.

```
!include /usr/share/kopano/ldap.openldap.cfg
#!include /usr/share/kopano/ldap.active-directory.cfg
```

Or on Kopano versions < 8.2:

```
$ sudo cd /etc/kopano/
$ sudo cp ldap.openldap.cfg ldap.cfg
```

For this howto we will be using anonymous binding

Edit /etc/kopano/ldap.cfg

Add your LDAP bind user and password if you do not use anonymous bind;

```
ldap_bind_user =
ldap_bind_password =
```

Change the search base so it matches our organisation

```
ldap_search_base = dc=example,dc=local
```

Restart the kopano-server

```
$ sudo /etc/init.d/kopano-server restart
```

Check if kopano can get the user from LDAP

```
$ sudo kopano-admin -l

User list for Default(2):
Username Fullname Homeserver
-----
SYSTEM SYSTEM kopano
john John Doe
```

Lets show the details of our user john.

```
$ sudo kopano-admin --details john

Username: john
Fullname: John Doe
Emailaddress: john@example.local
Active: yes
Administrator: yes
Address book: Visible
Auto-accept meeting req:no
Mapped properties:
  PR_SURNAME Doe
  PR_EC_ENABLED_FEATURES imap
  PR_EC_DISABLED_FEATURES pop3
Current user store quota settings:
  Quota overrides: yes
  Warning level: 953.67 MB
  Soft level: 1049.04 MB
  Hard level: 1144.41 MB
Current store size: 0.00 MB
Groups (1):
  Everyone
```

Ldap optimization

Create a file called optimize-index.ldif containing:

```
dn: olcDatabase={1}hdb,cn=config
changetype: modify
add: olcDbIndex
olcDbIndex: cn eq,sub
olcDbIndex: gidNumber eq
olcDbIndex: mail eq
olcDbIndex: memberUid eq
olcDbIndex: ou eq
olcDbIndex: uid eq
olcDbIndex: uidNumber eq
olcDbIndex: uniqueMember eq
olcDbIndex: kopanoAccount eq,pres
olcDbIndex: kopanoAliases eq
olcDbIndex: kopanoViewPrivilege eq
olcDbIndex: sn eq,sub
olcDbIndex: givenName eq,sub
```

*note depending on your choice during installation the hdb could be mdb,

Add the ldif to add the new indexes.

```
$ sudo cat optimize-index.ldif | ldapmodify -Y EXTERNAL -H ldapi:///
```

Check if our new olcDbIndex keys have been added.

```
$ sudo slapcat -b cn=config | grep olcDbIndex:
olcDbIndex: cn eq,sub
olcDbIndex: gidNumber eq
olcDbIndex: mail eq
olcDbIndex: memberUid eq
olcDbIndex: ou eq
olcDbIndex: uid eq
olcDbIndex: uidNumber eq
olcDbIndex: uniqueMember eq
olcDbIndex: kopanoAccount eq,pres
olcDbIndex: kopanoAliases eq
olcDbIndex: kopanoViewPrivilege eq
olcDbIndex: sn eq,sub
olcDbIndex: givenName eq,sub
```

You could check your slapd logging for suggestion of additional candidates for indexation.

```
$ sudo cat /var/log/syslog |grep bdb_equality_candidates
```

Ldap backup and restore using slapcat / slapadd

Backup

For the configuration use the the 0 since it is the first database.

```
$ sudo slapcat -n 0 -l config.ldif
```

For the organisation use the the 1 since it is the second database.

```
$ sudo slapcat -n 1 -l example.local.ldif
```

Restore

Make sure you have stopped slapd before doing this.

You can use slapadd -n 0/1 to restore the respective databases.

```
$ sudo slapadd -n 0 -l config.ldif
$ sudo slapadd -n 1 -l example.local.ldif
```

Be careful to check if your restored databases end up in `/var/lib/ldap` with the correct permissions.

The owner should be `openldap:openldap` and the permissions `0600`

Disable anonymous binding

If required you can disable anonymous binding.

Taken from <http://serverfault.com/questions/325912/disallow-global-anonymous-bind-with-cn-config>

Changing the default behaviour

Create a file `disable_anon_backend.ldif`

```
dn: olcDatabase={1}hdb,cn=config
add: olcRequires
olcRequires: authc
```

*note depending on your choice during installation the hdb could be mdb.

Create a file `disable_anon_frontend.ldif`

```
dn: olcDatabase={-1}frontend,cn=config
add: olcRequires
olcRequires: authc
```

Use `ldapmodify` to commit these changes.

```
$ sudo ldapmodify -Y EXTERNAL -H ldapi:/// -f disable_anon_frontend.ldif
$ sudo ldapmodify -Y EXTERNAL -H ldapi:/// -f disable_anon_backend.ldif
```

Testing if it works

After this the following should not be possible anymore.

```
$ sudo ldapsearch -xLLL -b dc=example,dc=local uid=john
```

The following should work.

```
$ sudo ldapsearch -x -D cn=admin,dc=example,dc=local -W -b
dc=example,dc=local
```

Modify `kopano ldap.cfg`


```
ldap_bind_user = cn=admin,dc=example,dc=local  
ldap_bind_passwd = writtendownearlier
```

Restart the kopano server

```
$ sudo service kopano-server restart
```

Check kopano users list

```
$ sudo kopano-admin -l
```

Related articles

- [Configure repositories](#)

```
$ sudo ldapadd -x -D cn=admin,dc=example,dc=local -W -f user.ldif
```