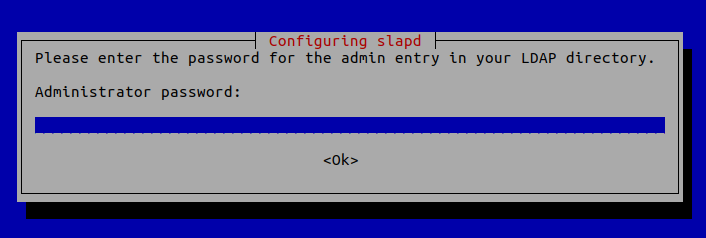
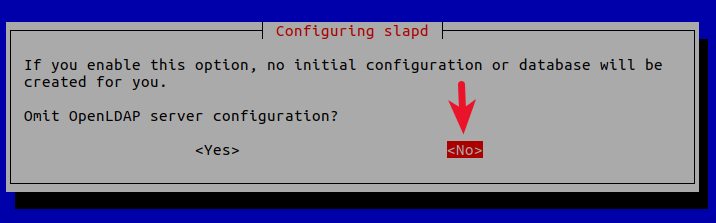
How to create LDAPServer

1. Create Ubuntu Instance 16.04(LTS SSD) in AWS
2. Login in as Ubuntu (replacing Ec2-user)
3. By using command “*sudo apt install slapd ldap-utils*”
4. It will be asked to set a password for the admin entry in the LDAP Directory
5. 
6. You can choose any password (eg. 123456)
7. Once it’s done, slapd will be automatically started. You can check out its status with: “*systemctl status slapd”*
8. By default, it runs as the openldap user as is defined in /etc/default/slapd file.
9. Run the following command to start the configuration wizard.

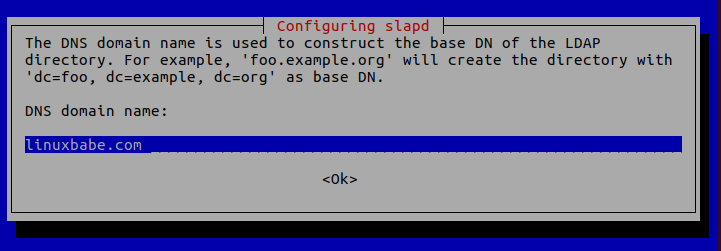
*“sudo dpkg-reconfigure slapd”*

1. You will need to answer a series of questions. Answer these questions as follows:

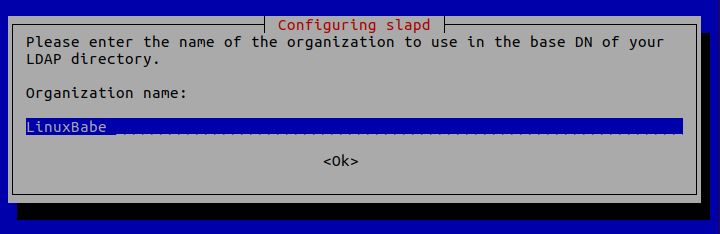
Omit LDAP server configuration: **NO**.



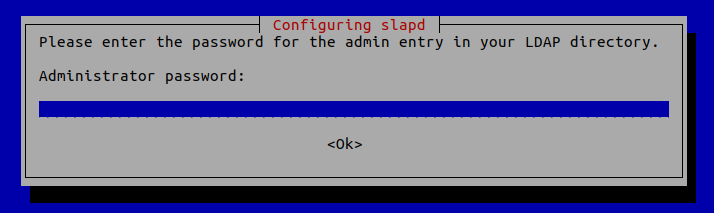
1. DNS domain name: Enter your domain name like **linuxbabe.com.** You will need to set a correct A record for your domain name.(for example we used linuxbabe.com)
2. This information is used to create the **base DN (distinguished name)** of the LDAP directory.



1. Organization name: Enter your organization name like **LinuxBabe.**



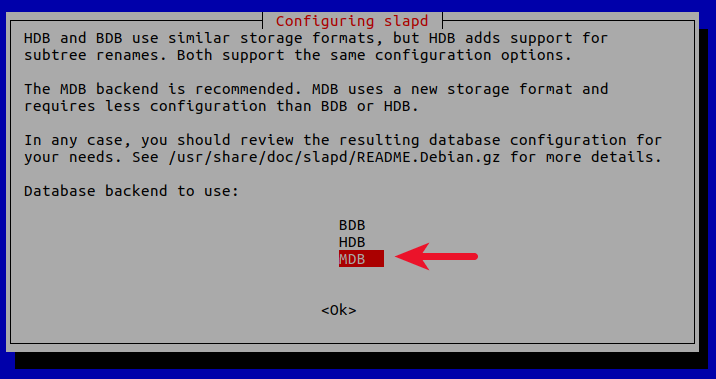
1. Administrator password: Enter the same password set during installation. (123456)



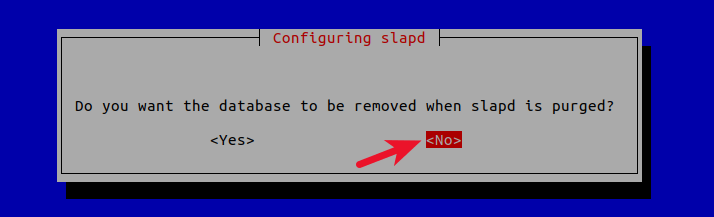
Database backend: **MDB**.

**BDB** (Berkeley Database) is slow and cumbersome. It is deprecated and support will be dropped in future OpenLDAP releases. **HDB** (Hierarchical Database) is a variant of the BDB backend and will also be deprecated.

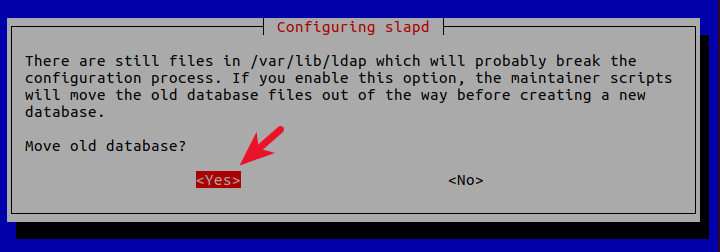
**MDB** reads are 5-20x faster than BDB. Writes are 2-5x faster. And it consumes 1/4 as much RAM as BDB. So we choose MDB as the database backend.



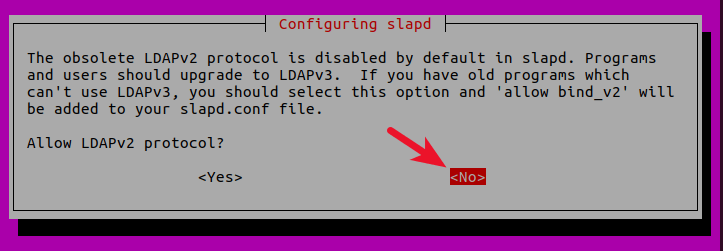
Do you want the database to be removed when slapd is purged? **No**.



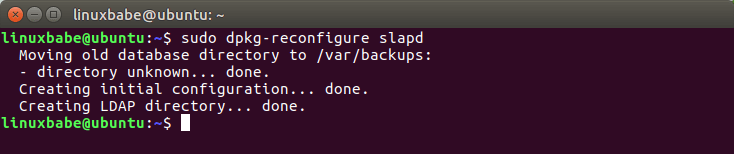
Move old database? **Yes**.



Allow LDAPv2 protocol? **No**. The latest version of LDAP is LDAP v.3, developed in 1997. LDAPv2 is obsolete.



Now the process will reconfigure the OpenLDAP service according to your answers. Your OpenLDAP server is now ready to use.



**Configuring the LDAP Clients**

/etc/ldap/ldap.conf is the configuration file for all OpenLDAP clients. Open this file.

sudo nano /etc/ldap/ldap.conf

We need to specify two parameters: the **base DN** and the **URI** of our OpenLDAP server. Copy and paste the following text at the end of the file. Replace your-domain and com as appropriate.

BASE     dc=your-domain,dc=com

URI      ldap://localhost

The first line defines the base DN. It tells the client programs where to start their search in the directory. If you used a subdomain when configuring OpenLDAP server, then you need to add the subdomain here like so

BASE      dc=subdomain,dc=your-domain,dc=com

The second line defines the URI of our OpenLDAP server. Since the LDAP server and client are on the same machine, we should set the URI to ldap://localhost.

## Testing OpenLDAP Server

Now that OpenLDAP server is running and client configuration is done, run the following command to make test connections to the server.

ldapsearch -x

Output:

# extended LDIF

#

# LDAPv3

# base <dc=linuxbabe,dc=com> (default) with scope subtree

# filter: (objectclass=\*)

# requesting: ALL

#

# linuxbabe.com

dn: dc=linuxbabe,dc=com

objectClass: top

objectClass: dcObject

objectClass: organization

o: LinuxBabe

# admin, linuxbabe.com

dn: cn=admin,dc=linuxbabe,dc=com

objectClass: simpleSecurityObject

objectClass: organizationalRole

cn: admin

description: LDAP administrator

# search result

search: 2

**result: 0 Success**

# numResponses: 3

# numEntries: 2

**Result: 0 Success** indicates that OpenLDAP server is working. If you get the following line, then it’s not working.

result: 32 No such object

## Installing phpLDAPadmin

phpLDAPadmin is a web-based program for managing OpenLDAP server. The command-line utilities can be used to manage our OpenLDAP server, but for those who want an easy-to-use interface, you can install phpLDAPadmin.

Run the following command to install phpLDAPadmin from Ubuntu package repository.

sudo apt install phpldapadmin

If your Ubuntu server doesn’t have a web server running, then the above command will install the Apache web server as a dependency. If there’s already a web server such as [Nginx](https://www.linuxbabe.com/nginx), then Apache won’t be installed.

## If you use Apache

The installation will put a configuration file phpldapadmin.conf under /etc/apache2/conf-enabled/ directory. Once the installation is done, you can access phpLDAPadmin web interface at

your-server-ip/phpldapadmin

or

your-domain.com/phpldapadmin

To enable HTTPS, you can obtain and install a free TLS certificate issued from Let’s Encrypt.

## If you use Nginx

Nginx users will need to manually create a server block file for phpLDAPadmin.

sudo nano /etc/nginx/conf.d/phpldapadmin.conf

Copy the following text and paste it to the file. Replace ldap.your-domain.com with your preferred domain name.

server {

listen 80;

server\_name ldap.your-domain.com;

root /usr/share/phpldapadmin/htdocs;

index index.php index.html index.htm;

error\_log /var/log/nginx/phpldapadmin.error;

access\_log /var/log/nginx/phpldapadmin.access;

location ~ \.php$ {

fastcgi\_pass unix:/run/php/php7.0-fpm.sock;

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME $document\_root/$fastcgi\_script\_name;

include fastcgi\_params;

}

}

Save and close the file. Then text Nginx configurations.

sudo nginx -t

If the test is successful, reload Nginx for the changes to take effect.

sudo systemctl reload nginx

Now you can access phpLDAPadmin web interface at ldap.your-domain.com. To enable HTTPS, you can obtain and install a free TLS certificate issued from Let’s Encrypt.

## Configuring phpLDAPadmin

We need to do some configurations just like we did with the command-line client. The phpLDAPadmin configuration file is at /etc/phpldapadmin/config.php .

sudo nano /etc/phpldapadmin/config.php

Since OpenLDAP and phpLDAPadmin are running on the same machine, so we will configure phpLDAPadmin to connect to localhost on the default LDAP port 389 without SSL/TLS encryption.

Line 293 specifies that phpLDAPadmin will connect to localhost.

$servers->setValue('server','host','127.0.0.1');

Line 296 is commented out by default, which means the standard port 389 will be used.

// $servers->setValue('server','port',389);

Line 335 is commented out by default, which means TLS encryption is not enabled.

// $servers->setValue('server','tls',false);

Then go to line 300.

$servers->setValue('server','base',array('dc=example,dc=com'));

Change it to:

$servers->setValue('server','base',array());

This will let phpLDAPadmin automatically detect the base DN of your OpenLDAP server. Next, you can disable anonymous login. Go to line 453.

// $servers->setValue('login','anon\_bind',true);

By default, anonymous login is enabled. To disable it, you need to remove the comment character (the two slashes) and change true to false.

$servers->setValue('login','anon\_bind',false);

You will probably want to disable template warnings because these warnings are annoying and unimportant. Go to line 161.

// $config->custom->appearance['hide\_template\_warning'] = false;

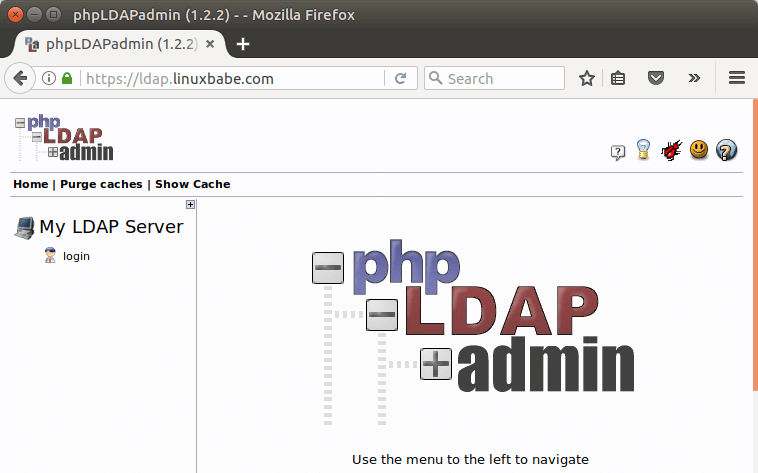
Remove the comment character and change false to true.

$config->custom->appearance['hide\_template\_warning'] = true;

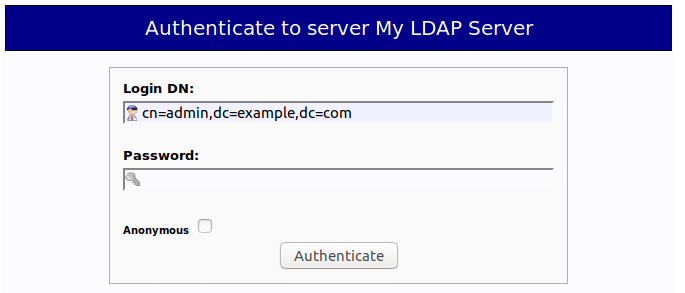
Save and close the file.

## Accessing phpLDAPadmin Web Interface

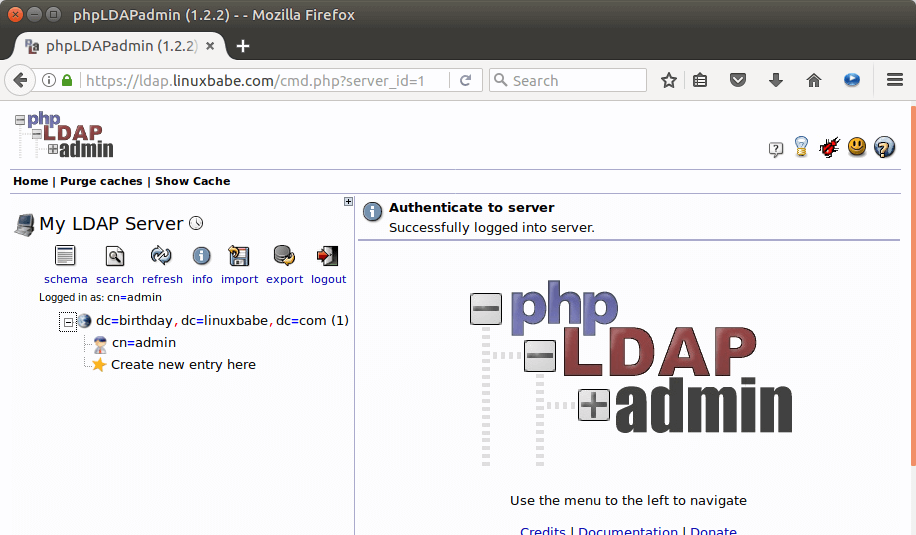
We can now test out the phpLDAPadmin tool with our web browser. When phpLDAPadmin first loads, it looks something like this.



To log into our OpenLDAP server, click on the login link. You will see the login dialog box. The default login DN is cn=admin,dc=example,dc=com. You may need to change dc=example. In my case, I need to change the login DN to cn=admin,dc=linuxbabe,dc=com.



The password is the admin password you set during the configuration of OpenLDAP server. Once you log into phpLDAPadmin, you can manage this directory server.



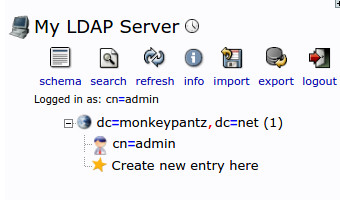
That’s it! I hope this tutorial helped you install and configure both **OpenLDAP server** and **phpLDAPadmin** on Ubuntu 16.04. In the next tutorial, we will see how to configure Ubuntu to authenticate user logins with OpenLDAP.

# **How to populate an LDAP server with users and groups via phpLDAPadmin**

## Creating Organizational Units

LDAP breaks everything into very specific pieces, and we're going to focus on two of those pieces: people and groups. Because we're creating fairly generic Organizational Units (OUs), we'll use the Generic Organizational Unit Template. To get there, log into phpLDAPadmin, click to expand your server listing (in my example it's dc=monkeypantz,dc=net) and then click Create New Entry Here (**Figure A**).

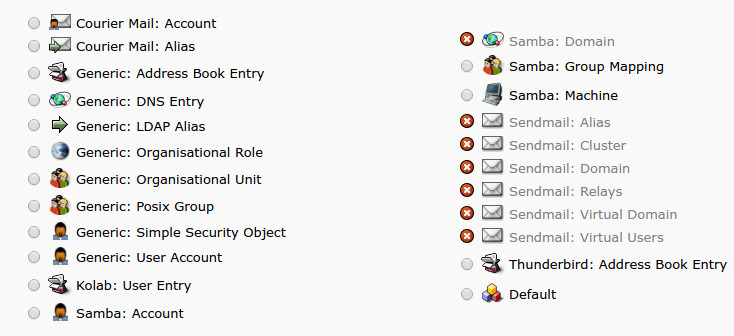
**Figure A**



Creating a new entry with phpLDAPadmin.

In the right pane (**Figure B**), select Generic: Organizational Unit.

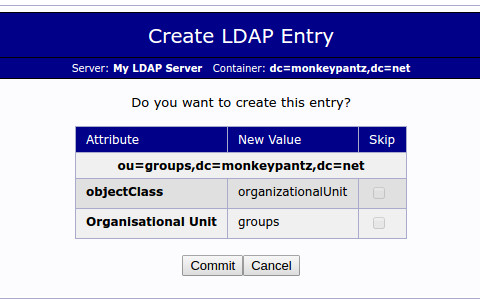
**Figure B**



Selecting from the many available objects on the LDAP server.

Let's first create an OU named "groups". In the next window type groups and click Create Object. Commit the group by clicking Commit in the next window (**Figure C**).

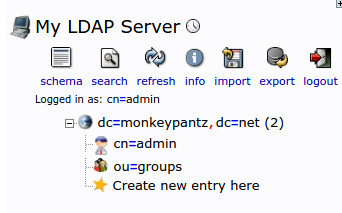
**Figure C**



Committing your changes to LDAP.

You'll see a new entry in the left pane called ou=groups (**Figure D**).

**Figure D**



Our new OU ready to be used.

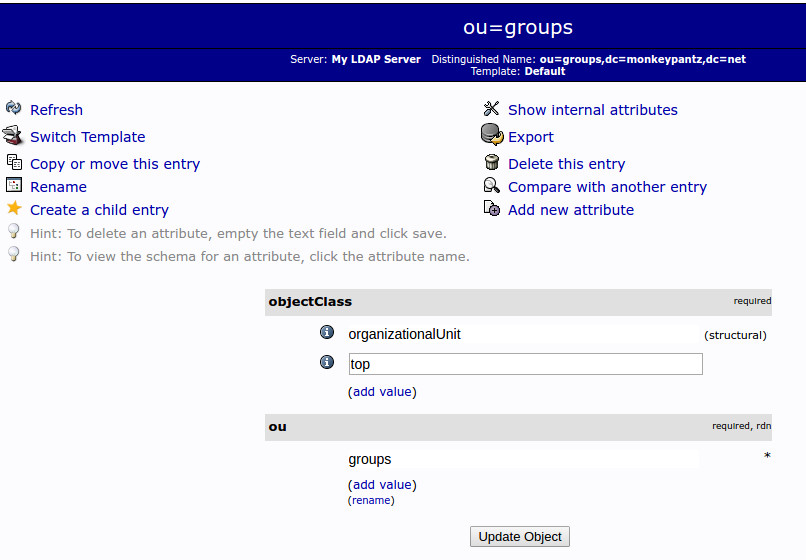
Let's create a new OU named "users". Walk through the same process as above, though name the OU "users" instead of "groups". You'll see "ou=groups" and "ou=users" in the left pane.

## Creating groups

Now that we have an OU created for groups, we can add the necessary groups. Let's create groups for "admin", "developers", and "users". Here's how.

1. Click the groups OU in the left pane.
2. In the resulting window, click Create Child Entry (**Figure E**).
3. Click Generic: Posix Group.
4. Type admin into the group text area.
5. Click Create Object.
6. Click Commit.
7. Repeat the process for "developers" and "users".

**Figure E**



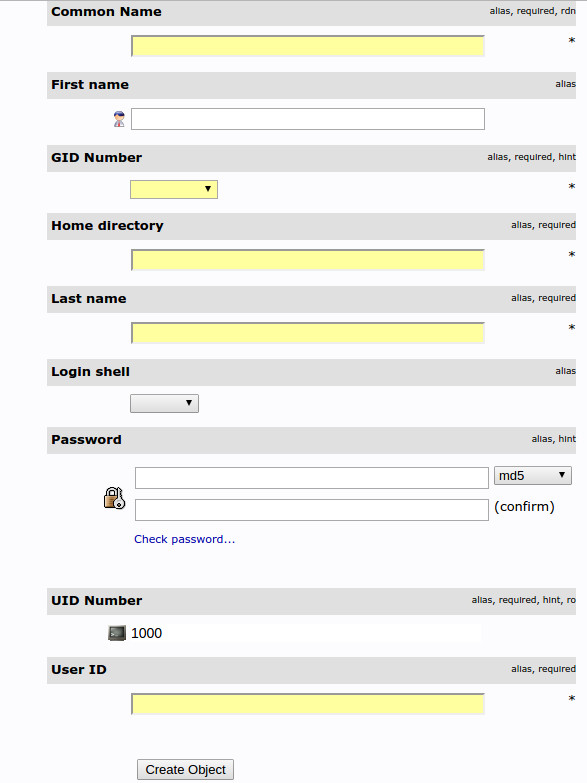
Creating a new group under the groups OU.

## Creating users

Now that we have our groups created, we want to create users. To do this, follow these steps.

1. Click ou=users from the left pane.
2. In the resulting window, click Create A Child Entry.
3. Select Generic: User Account.
4. Fill out the required information- note that Common Name must be unique (**Figure F**).
5. Click Create Object.
6. Click Commit.
7. Repeat this process until you have added your necessary users added.

**Figure F**



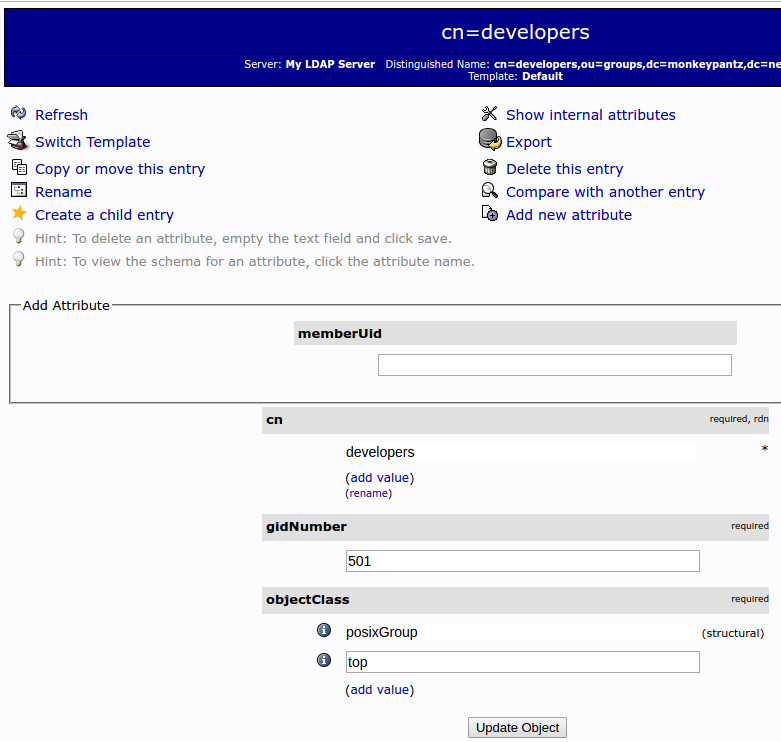
Creating a new user.

## Adding users to groups

To add a user to a group, you must know the user's UID (named User ID in the user creation window). To find a UID go to ou=users | View X child (where X is the number of users) and then locate the user to be added and make note of their associated UID. Once you have that UID, let's add that user to the developers group. Here's how.

1. Expand ou=groups.
2. Click the developers group.
3. Click Add New Attribute.
4. From the drop-down, select memberUID.
5. Enter the UID for the user in the memberUID section (**Figure G**).
6. Click Update Object.

**Figure G**



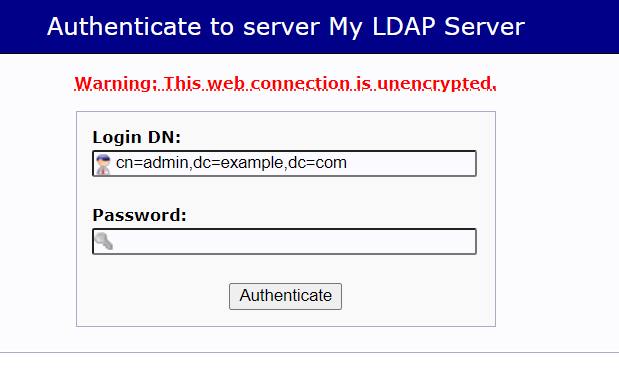
Adding a user to a group.

After you add the first user, adding subsequent users is much simpler. If you click the group name (under ou=groups in the left pane), you can click Modify Group Members (under memberUID) and then add the users from a list.

**Adding users and groups to Jenkins**

Login in Web browser as your ubuntu public ip and ://phpldapadmin (18.191.67.191://phpldapadmin)

Click on Login



Integrating LDAP servers with Jenkins:

Now,Login to Jenkins by using admin credentials.

Go to Manage Jenkins>Configure Golabal Security>Select LDAP in security realm and enter the server details in the following format.

Specify the name of the LDAP server host name(s) (like ldap.sun.com). If your LDAP server uses a port other than 389 (which is the standard for LDAP), you can also append a port number here, like ldap.acme.org:1389. To connect to LDAP over SSL (AKA LDAPS), specify it with the ldaps:// protocol, like ldaps://ldap.acme.org or ldaps://ldap.acme.org:1636 (if the port is other than the default 636).

The default port for LDAP servers is 389.So enter the LDAP server details as

ldap://public IP of the server: port number (default is 389)

Eg: ldap://18.191.67.191:389

Enter root DN with your domain name details

Eg: DN: dc=example,dc=com

Use filter search= cn{0}

Now test LDAP server. They are probably integrated with Jenkins by now.

Now go to Authorisation and select Matrix Based Security

Add users from LDAP servers and assign permissions as per the requirements.

Apply and save. These users can interact with Jenkins with their credentials as far as their permissions go.

Done.