Practical 7:- Configure LDAP Server, Configure LDAP Server in order to share users' accounts in your local networks, Add LDAP User Accounts in the OpenLDAP Server, Configure LDAP Client in order to share users' accounts in your local networks. Install phpLDAPadmin to operate LDAP server via Web browser.

Step 1:- Install OpenLDAP Server

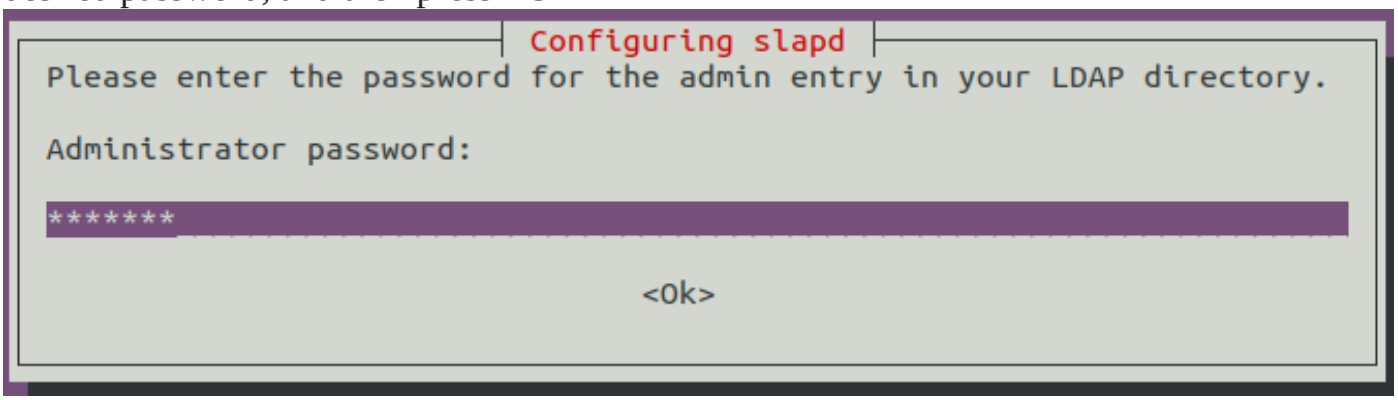
Install OpenLDAP and its utilities using apt-get and enable it during start-up. While installing, it will ask to provide admin password.

 # sudo apt-get update
 # apt-get install slapd ldap-utils
 # systemctl enable slapd

When done, install LDAP packages by running the commands below:

```
rootclient@ubuntu:~$ sudo apt-get -y install slapd ldap-utils
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libodbc1
Suggested packages:
  libsasl2-modules-gssapi-mit | libsasl2-modules-gssapi-heimdal libmyodbc
  odbc-postgresql tdsodbc unixodbc-bin
The following NEW packages will be installed:
  ldap-utils libodbc1 slapd
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 1,692 kB of archives.
After this operation, 17.1 MB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu bionic/main amd64 libodbc1 amd64 2.3.4-1.1ubuntu
3 [183 kB]
```

During the installation, you'll be prompted to set LDAP **admin password**, provide your desired password, and then press <**OK**>



Confirm the password and continue installation by selecting **<ok>** with TAB key.

```
Configuring slapd
Please enter the admin password for your LDAP directory again to verify
that you have typed it correctly.

Confirm password:

*******

<Ok>
```

```
rootclient@ubuntu:~$ sudo systemctl start slapd
rootclient@ubuntu:~$ sudo systemctl status slapd
● slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol)
   Loaded: loaded (/etc/init.d/slapd; generated)
  Drop-In: /lib/systemd/system/slapd.service.d
           └─slapd-remain-after-exit.conf
   Active: active (running) since Sun 2019-09-22 08:00:38 PDT; 14min ago
     Docs: man:systemd-sysv-generator(8)
    Tasks: 3 (limit: 4668)
   CGroup: /system.slice/slapd.service
           └─5171 /usr/sbin/slapd -h ldap:/// ldapi:/// -g openldap -u openldap -F /etc/ld

Sep 22 08:00:38 google.com systemd[1]: Starting LSB: OpenLDAP standalone server (Lightweig
Sep 22 08:00:38 google.com slapd[5164]:  * Starting OpenLDAP slapd
Sep 22 08:00:38 google.com slapd[5170]: @(#) $OpenLDAP: slapd  (Ubuntu) (Aug  8 2019 18:08
                                                Debian OpenLDAP Maintainers <pkg-openldap-
Sep 22 08:00:38 google.com slapd[5171]: slapd starting
Sep 22 08:00:38 google.com slapd[5164]:    ...done.
Sep 22 08:00:38 google.com systemd[1]: Started LSB: OpenLDAP standalone server (Lightweigh
```

Using netstat, check if the slapd is running in the port no 5171

 # netstat –pltn

```
rootclient@ubuntu:~$ sudo netstat -pltn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN      430/rpcbind
tcp        0      0 0.0.0.0:627             0.0.0.0:*               LISTEN      874/rpc.ypxfrd
tcp        0      0 192.168.171.133:53      0.0.0.0:*               LISTEN      631/named
tcp        0      0 127.0.0.1:53            0.0.0.0:*               LISTEN      631/named
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      432/systemd-resolve
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      2319/cupsd
tcp        0      0 127.0.0.1:953           0.0.0.0:*               LISTEN      631/named
tcp        0      0 0.0.0.0:445             0.0.0.0:*               LISTEN      920/smbd
tcp        0      0 0.0.0.0:638             0.0.0.0:*               LISTEN      885/ypbind
tcp        0      0 0.0.0.0:610             0.0.0.0:*               LISTEN      857/ypserv
tcp        0      0 0.0.0.0:389             0.0.0.0:*               LISTEN      5171/slapd
tcp        0      0 0.0.0.0:139             0.0.0.0:*               LISTEN      920/smbd
tcp6       0      0 :::111                  :::*                    LISTEN      430/rpcbind
tcp6       0      0 ::1:631                 :::*                    LISTEN      2319/cupsd
tcp6       0      0 :::445                  :::*                    LISTEN      920/smbd
tcp6       0      0 :::389                  :::*                    LISTEN      5171/slapd
tcp6       0      0 :::139                  :::*                    LISTEN      920/smbd
```

The OpenLDAP package have been installed and now we are going to reconfigure all the defaults those are shipped with ubuntu. Execute the following command to bring up package configuration tool.

 # sudo dpkg-reconfigure slapd

The package configuration tool will ask a series of question for re-configuring OpenLDAP

→Omit OpenLDAP server configuration? <No>

→DNS domain name: rizviclient.com

→Organization name: rizvi

→Enter password and confirm it: password

→Database backend to use: HDB

→Do you want the database to be removed when slapd is purged? <No>

→Move old database? <Yes>

→Allow LDAPv2 protocol? <No>

```
rootclient@google:~$ sudo dpkg-reconfigure slapd
[sudo] password for rootclient:
  Backing up /etc/ldap/slapd.d in /var/backups/slapd-2.4.45+dfsg-1ubuntu1.4... d
one.
  Moving old database directory to /var/backups:
  - directory unknown... done.
  Creating initial configuration... done.
  Creating LDAP directory... done.
rootclient@google:~$
```

Restart OpenLDAP

 # sudo systemctl restart slapd

At this stage, we have installed and reconfigured OpenLDAP server.

At this point, your LDAP server is configure and running. Open up the LDAP port on your firewall so external clients can connect.

#sudo ufw allow ldap

Let's test your LDAP connection with **ldapwhoami** command, which should return the user name we're connected as:-

 # sudo ldapwhoami –H ldap:// -x

```
rootclient@google:~$ sudo ufw allow ldap
Rules updated
Rules updated (v6)
rootclient@google:~$ sudo ldapwhoami -H ldap:// -x
anonymous
```

Anonymous

Anonymous is the result we're expecting, since we ran ldapwhoami without logging in the LDAP server. This means the server is running and answering queries. Next we'll setup a web interface to manage LDAP data.

Step 2:- installing and configuring the phpLDAPadmin web interface

For created/edited/searched OU, groups, users through command line. However you can do the same using a web interface called phpldapadmin. The phpldapadmin is shipped along with ubuntu by default. Use apt-get to install it.

 # sudo apt-get install phpldapadmin

```
rootclient@google:~$ sudo apt-get install phpldapadmin
[sudo] password for rootclient:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapache2-mod-php7.2 libapr1
  libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0 php
  php-common php-ldap php-xml php7.2 php7.2-cli php7.2-common php7.2-json
  php7.2-ldap php7.2-opcache php7.2-readline php7.2-xml
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom php-pear
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapache2-mod-php7.2 libapr1
  libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0 php
  php-common php-ldap php-xml php7.2 php7.2-cli php7.2-common php7.2-json
  php7.2-ldap php7.2-opcache php7.2-readline php7.2-xml phpldapadmin
0 upgraded, 23 newly installed, 0 to remove and 0 not upgraded.
Need to get 6,435 kB of archives.
After this operation, 29.7 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu bionic/main amd64 libapr1 amd64 1.6.3-
2 [90.9 kB]
```

Edit the config file for phpldapadmin to reflect the directory structure that we have created earlier.

#sudo nano /etc/phpldapadmin/config.php

```
  GNU nano 2.9.3              /etc/phpldapadmin/config.php              Modified

$servers->setValue('server','name','My LDAP Server');

/* Examples:
   'ldap.example.com',
   'ldaps://ldap.example.com/',
   'ldapi://%2fusr%local%2fvar%2frun%2fldapi'
           (Unix socket at /usr/local/var/run/ldap) */
$servers->setValue('server','host','192.168.171.133');

/* The port your LDAP server listens on (no quotes). 389 is standard. */
$servers->setValue('server','port',389);

/* Array of base DNs of your LDAP server. Leave this blank to have phpLDAPadmin
   auto-detect it for you. */
$servers->setValue('server','base',array('dc=rizvi,dc=com'));

/* Five options for auth_type:
   1. 'cookie': you will login via a web form, and a client-side cookie will
      store your login dn and password.
   2. 'session': same as cookie but your login dn and password are stored on the
      web server in a persistent session variable.
   3. 'http': same as session but your login dn and password are retrieved via
      HTTP authentication.
   4. 'config': specify your login dn and password here in this config file. No
      login will be required to use phpLDAPadmin for this server.
   5. 'sasl': login will be taken from the webserver's kerberos authentication.
      Currently only GSSAPI has been tested (using mod_auth_kerb).

   Choose wisely to protect your authentication information appropriately for
   your situation. If you choose 'cookie', your cookie contents will be
   encrypted using blowfish and the secret your specify above as
   session['blowfish']. */
$servers->setValue('login','auth_type','session');

/* The DN of the user for phpLDAPadmin to bind with. For anonymous binds or
   'cookie','session' or 'sasl' auth_types, LEAVE THE LOGIN_DN AND LOGIN_PASS
   BLANK. If you specify a login_attr in conjunction with a cookie or session
```

You can now access phpldapadmin through http://LDAP-SERVER-IP/phpldapadmin. Login with user as default directory structure and password as 'password'. To password protect the phpldapadmin location, create an user using apache utils **htpasswd**.

 # sudo htpasswd -c /etc/apache2/htpasswd ldapadminuser

```
rootclient@google:~$ sudo htpasswd -c /etc/apache2/htpasswd ldapadminuser
New password:
Re-type new password:
Adding password for user ldapadminuser
```

Append the following section in apache's main configuration file **/etc/apache2/apache2.conf**

 # sudo nano /etc/apache2/apache2.conf

```
  GNU nano 2.9.3                /etc/apache2/apache2.conf

# Use mod_remoteip instead.
#
LogFormat "%v:%p %h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\""
LogFormat "%h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" combi
LogFormat "%h %l %u %t \"%r\" %>s %O" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.

# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf

# Include the virtual host configurations:
IncludeOptional sites-enabled/*.conf

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
<Location /phpldapadmin>
AuthType Basic
AuthName "Restricted Files"
AuthUserFile /etc/apache2/htpasswd
Require valid-user
</Location>
```

Restart Apache

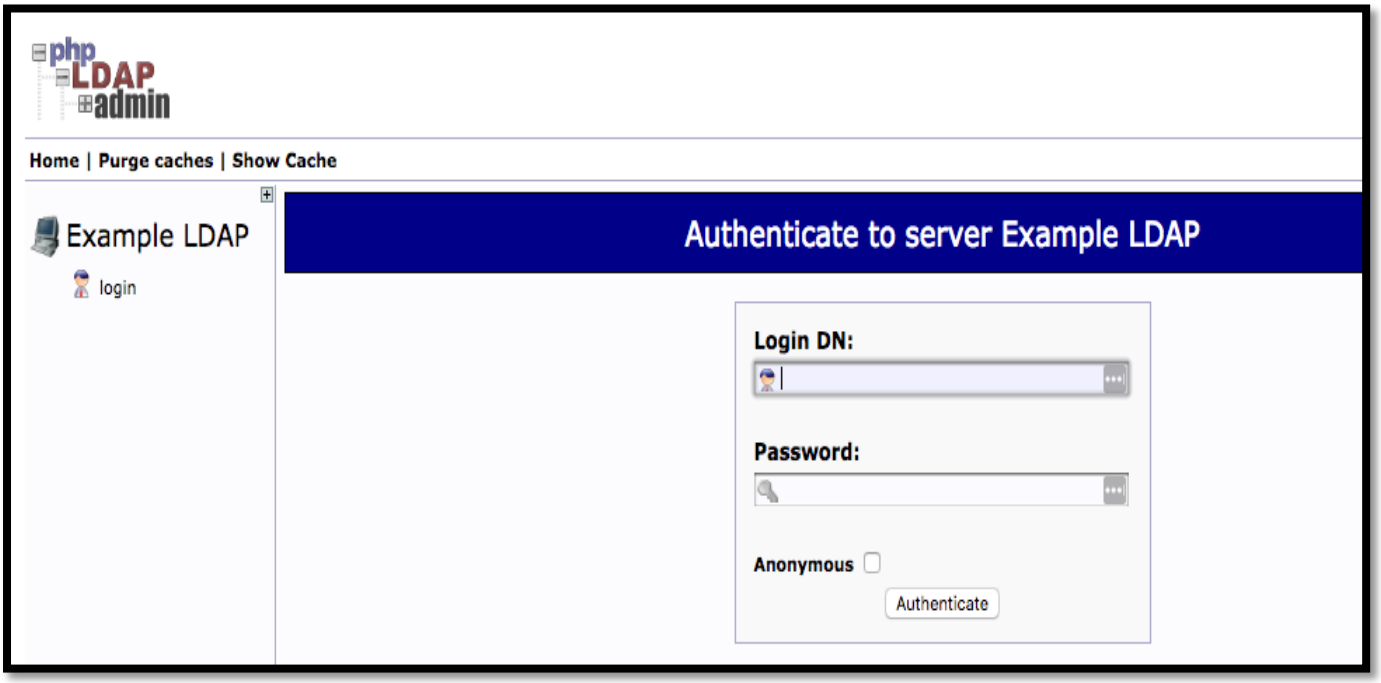
 # sudo systemctl restart apache2

Step 3:- Logging into phpldapadmin web Interface

Navigate to the application in your web browser

https://example.com/phpldapadmin

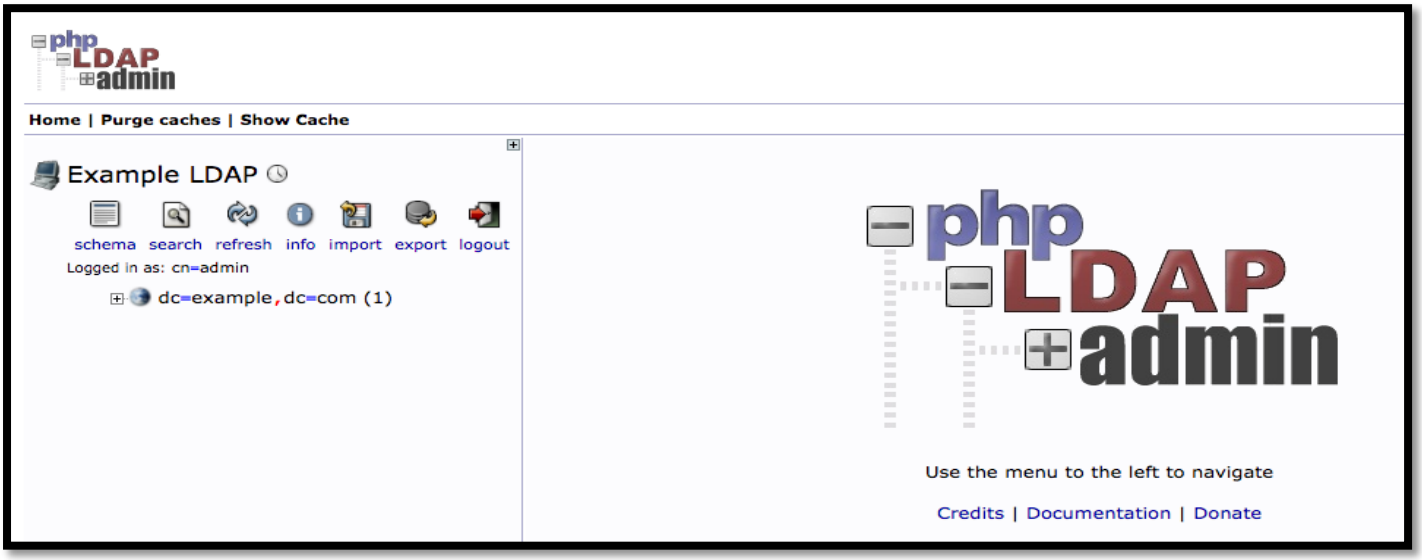
Refresh the phpldapadmin page, you will see the password prompt that you configured using htpasswd utils. The phpLDAPadmin landing page will load. Click on the login link in the left-hand menu on the page. A login form will be presented:



The Login DN is the username that you will be using. It contains the account name as a cn= section, and the domain name you selected for the server broken into dc= sections as described in previous steps. The default admin account that we set up during install is called admin, so for our example we would type in the following:

cn=admin,dc=rizviclient ,dc=com

After entering the appropriate string for your domain, type in the admin password you created during configuration, then click the Authenticate button.
You will be taken to the main interface:



At this point, you are logged into the phpLDAPadmin interface. You have the ability to add users, organizational units, groups, and relationships.

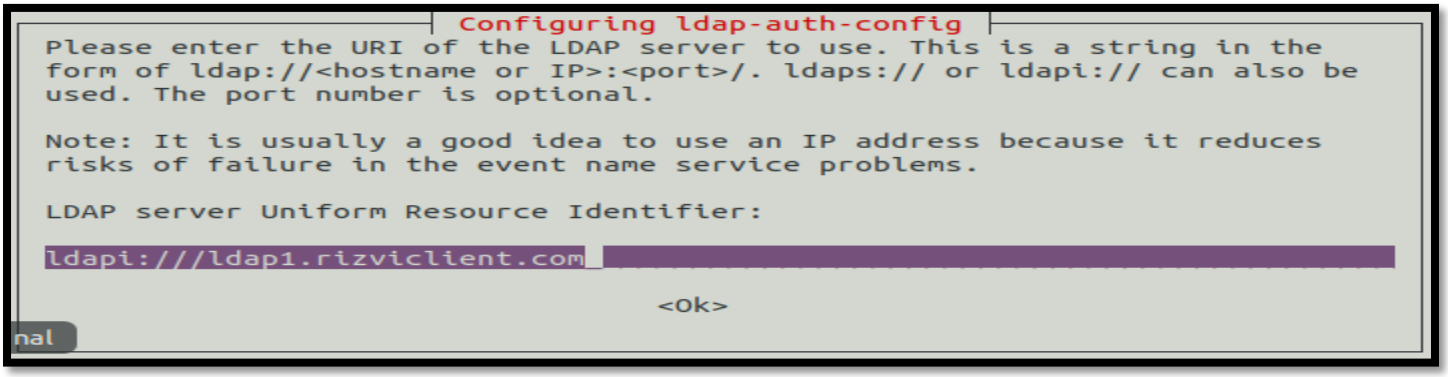
Step 4:- Configure LDAP Client

First start by installing the necessary packages by running the following command.

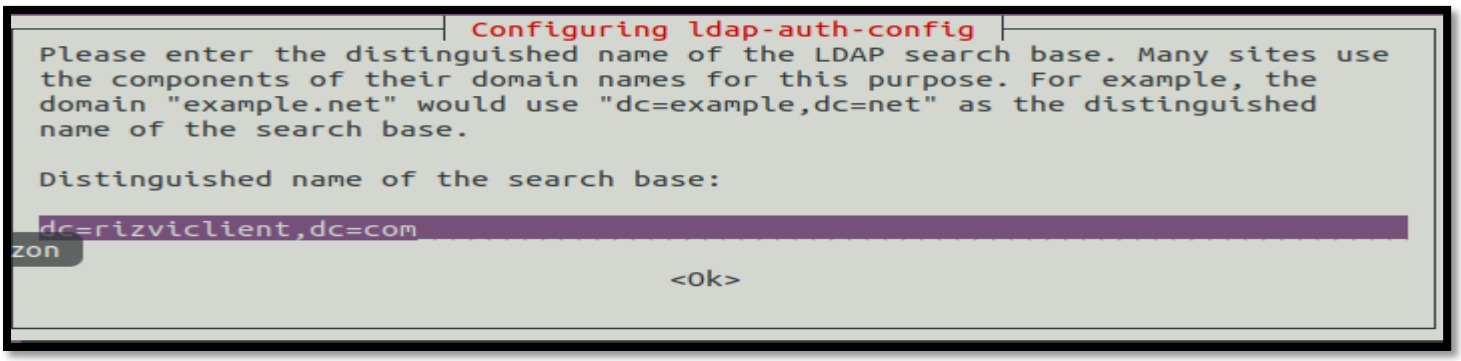
#sudo apt-get install libnss-ldap libpam-ldap ldap-utils nscd

```
rootclient@google:~$ sudo apt-get install libnss-ldap libpam-ldap ldap-utils nscd
[sudo] password for rootclient:
Reading package lists... Done
Building dependency tree
Reading state information... Done
ldap-utils is already the newest version (2.4.45+dfsg-1ubuntu1.4).
The following additional packages will be installed:
  auth-client-config ldap-auth-client ldap-auth-config
Suggested packages:
  libpam-cracklib
The following NEW packages will be installed:
  auth-client-config ldap-auth-client ldap-auth-config libnss-ldap libpam-ldap
  nscd
0 upgraded, 6 newly installed, 0 to remove and 0 not upgraded.
Need to get 203 kB of archives.
After this operation, 951 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

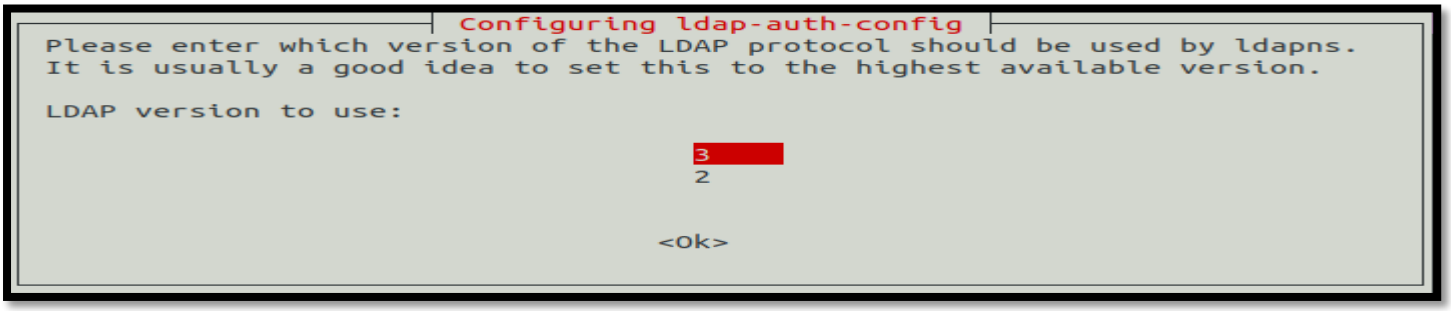
During the installation, you will be prompted for details of your LDAP server (provide the values according to your environment). Note that the ldap-auth-config package which is auto-installed does the most of the configurations based on the inputs you enter.



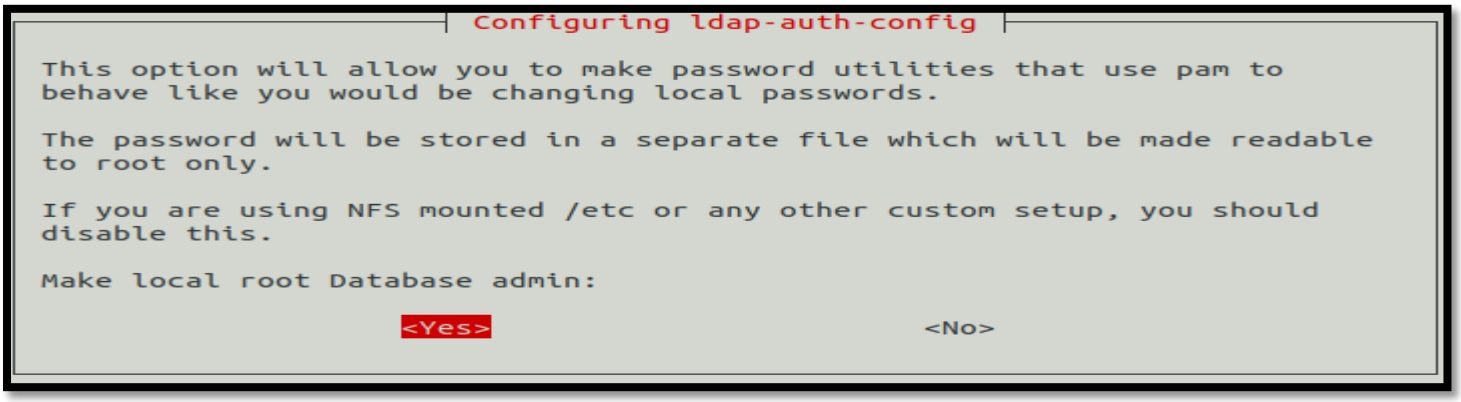
Next, enter the name of the LDAP search base, you can use the components of their domain names for this purpose as shown in the screenshot.



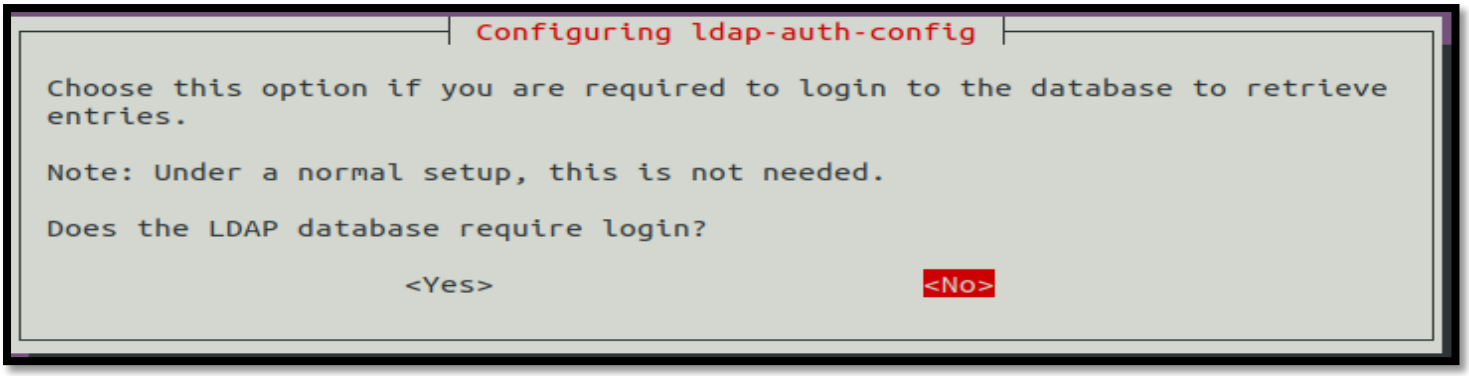
Also choose the LDAP version to use and click Ok.



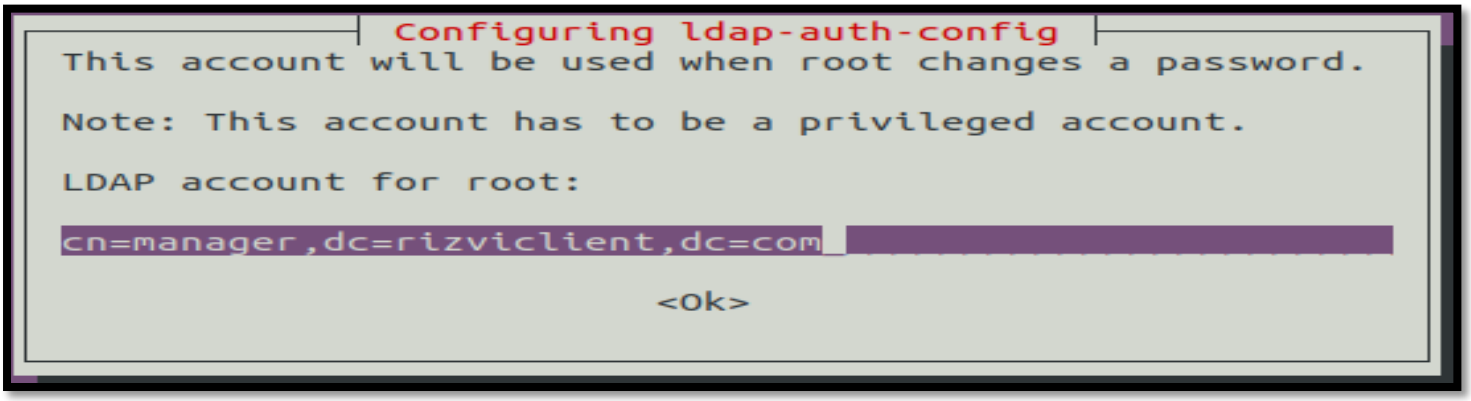
Now configure the option to allow you to make password utilities that use pam to behave like you would be changing local passwords and click Yes to continue..



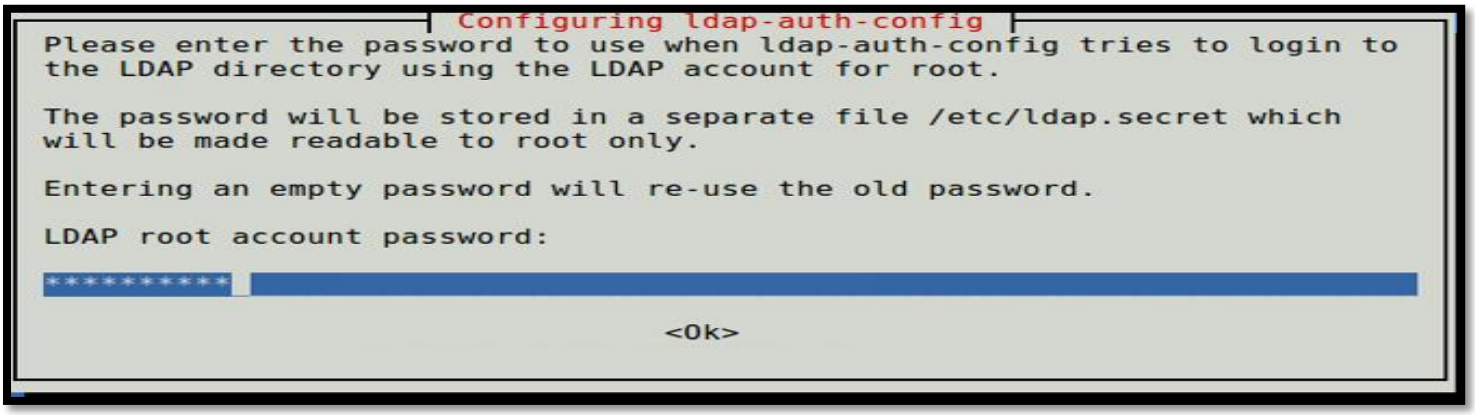
Next, disable login requirement to the LDAP database using the next option.



Disable Login to LDAP Database also define LDAP account for root and click Ok.



Define LDAP Account for Root Next, enter the password to use when ldap-auth-config tries to login to the LDAP directory using the LDAP account for root.



Enter LDAP Root Password

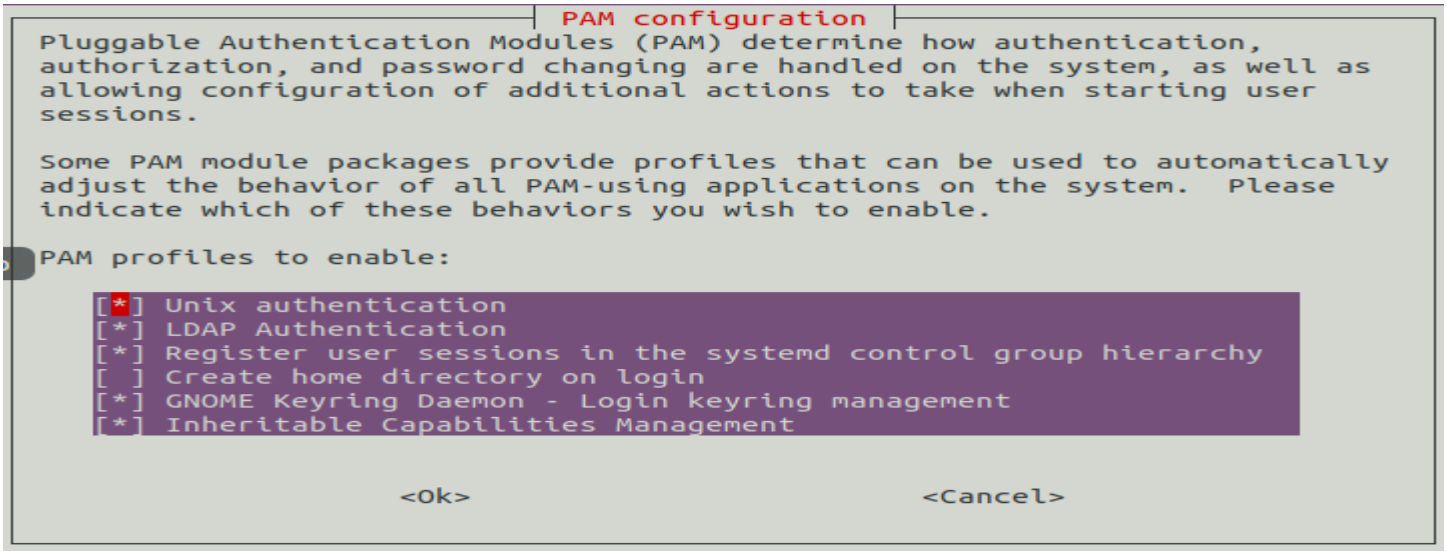
The results of the dialog will be stored in the file /etc/ldap.conf. If you want to make any alterations, open and edit this file using your favorite command line editor.

Next, configure the LDAP profile for NSS by running.

$ sudo auth-client-config -t nss -p lac_ldap

Then configure the system to use LDAP for authentication by updating PAM configurations. From the menu, choose LDAP and any other authentication mechanisms you need. You should now be able to log in using LDAP-based credentials.

$ sudo pam-auth-update



Configure PAM Authentication Mechanism

In case you want the home directory of the user to be created automatically, then you need to perform one more configuration in the common-session PAM file.

$ sudo nano /etc/pam.d/common-session

Add this line in it.

session required pam_mkhomedir.so skel=/etc/skel umask=077

Save the changes and close the file. Then restart the NCSD (Name Service Cache Daemon) service with the following command.

$ sudo systemctl restart nscd

$ sudo systemctl enable nscd