

Installation via Docker

Beschreibung:

Ein Idap Server für Adressbücher mit carddav sync.

Installation:

Docker installieren

```
apt install docker.io docker-compose curl
```

Nun Projektverzeichnisse erstellen

```
ldap-carddav-stack/  
├─ docker-compose.yml  
├─ .env  
├─ ldap-carddav/  
│ └─ conf.php # deine Konfiguration  
├─ Dockerfile # für ldap-carddav  
├─ ldap-carddav-data
```

```
mkdir -p /root/ldap-carddav-stack/ldap-carddav  
mkdir -p /root/ldap-carddav-stack/ldap-carddav-data
```

Dockerfile erstellen zum image bauen

```
nano /root/ldap-carddav-stack/Dockerfile
```

Inhalt

```
FROM debian:bullseye
```

```
ENV DEBIAN_FRONTEND=noninteractive
```

```
# Abhängigkeiten installieren
```

```
RUN apt-get update && apt-get install -y \
```

```
    apache2 \
```

```
    php \
```

```
    php-ldap \
```

```
    php-xml \
```

```
    php-mbstring \
```

```
    php-sqlite3 \
```

```
    sqlite3 \
```

```
    libapache2-mod-php \
```

```
    nano \
```

```
    curl \
```

```
    ldap-utils \
```

```
    composer \
```

```
    && apt-get clean
```

```
# ldap-carddav klonen
```

```
RUN git clone https://github.com/isubsoft/ldap-carddav.git /var/www/html/ldap-carddav
```

```
# Composer-Abhängigkeiten installieren
```

```
WORKDIR /var/www/html/ldap-carddav
```

```
RUN composer install
```

```
# Rewrite-Modul aktivieren
```

```
RUN a2enmod rewrite
```

```
# 000-default.conf ersetzen
```

```
RUN rm /etc/apache2/sites-enabled/000-default.conf && \
```

```
    echo '<VirtualHost *:80>\n\
```

```
    ServerAdmin admin@example.org\n\
```

```
    DocumentRoot /var/www/html/ldap-carddav\n\
```

```
\n\
```

```
    <Directory /var/www/html/ldap-carddav>\n\
```

```
        Options Indexes FollowSymLinks\n\
```

```
        AllowOverride All\n\
```

```
        Require all granted\n\
```

```
DirectoryIndex server.php\n\nRewriteEngine On\n\nRewriteCond %{REQUEST_FILENAME} !-f\n\nRewriteCond %{REQUEST_FILENAME} !-d\n\nRewriteRule ^(.*)$ server.php [QSA,L]\n\n</Directory>\n\n\n\nRedirect 301 /.well-known/carddav /server.php\n\n</VirtualHost>' > /etc/apache2/sites-enabled/000-default.conf\n\n# Apache starten\nCMD ["apachectl", "-D", "FOREGROUND"]\n\nEXPOSE 80
```

Die .env Datei

```
nano /root/ldap-carddav-stack/.env
```

Inhalt

```
LDAP_ORGANISATION=ExampleCorp\nLDAP_DOMAIN=example\nLDAP_TOP_DOMAIN=local\nLDAP_ADMIN_PASSWORD=admin
```

Die compose Datei

```
nano /root/ldap-carddav-stack/docker-compose.yml
```

Inhalt

```
version: '3.8'\n\nservices:\n  ldap:
```

image: osixia/openldap:1.5.0

container_name: ldap

environment:

LDAP_ORGANISATION: \${LDAP_ORGANISATION}

LDAP_DOMAIN: \${LDAP_DOMAIN}.\${LDAP_TOP_DOMAIN}

LDAP_ADMIN_PASSWORD: \${LDAP_ADMIN_PASSWORD}

volumes:

- ./ldap_data:/var/lib/ldap

- ./ldap_config:/etc/ldap/slapd.d

ports:

- "389:389"

phpldapadmin:

image: osixia/phpldapadmin:0.9.0

container_name: phpldapadmin

environment:

PHPLDAPADMIN_LDAP_HOSTS: ldap

ports:

- "6443:443"

carddav:

build:

context: .

dockerfile: Dockerfile

container_name: ldap-carddav

ports:

- "80:80"

volumes:

- ./ldap-carddav/conf.php:/var/www/html/ldap-carddav/conf/conf.php:ro

- ./ldap-carddav-data:/var/www/html/ldap-carddav/data

depends_on:

- ldap

environment:

- LDAP_HOST=ldap

- LDAP_BASE_DN=dc=\${LDAP_DOMAIN},dc=\${LDAP_TOP_DOMAIN}

- LDAP_BIND_DN=cn=admin,dc=\${LDAP_DOMAIN},dc=\${LDAP_TOP_DOMAIN}

- LDAP_BIND_PASSWORD=\${LDAP_ADMIN_PASSWORD}

- LDAP_BASE_DN_SYNC=dc=\${LDAP_DOMAIN},dc=\${LDAP_TOP_DOMAIN}

- LDAP_BIND_DN_SYNC=cn=shacker,dc=\${LDAP_DOMAIN},dc=\${LDAP_TOP_DOMAIN}

- LDAP_BIND_PASSWORD_SYNC=1234

Container starten:

```
docker-compose up -d
```

PHP File

```
nano /root/ldap-carddav-stack/ldap-carddav/conf.php
```

Version 1: Nur ein Adressbuch, für alle schreibend:

Inhalt

```
<?php

$config = [];

// === TEMP / DATA ===
$config['tmpdir'] = '%systemtmpdir';
$config['datadir'] = '/var/www/html/ldap-carddav/data';

// === DATABASE ===
$config['sync_database'] = [
    'dsn' => 'sqlite:/var/www/html/ldap-carddav/data/cards.db',
    'username' => '',
    'password' => '',
    'options' => [],
    'init_commands' => []
];

// === LDAP SERVER CONFIG ===
$config['server']['ldap'] = [
    'host' => getenv('LDAP_HOST') ?: 'localhost',
    'network_timeout' => 10,
    'connection_security' => 'none'
];

// === LDAP AUTH ===
$config['auth']['ldap'] = [
    'base_dn' => getenv('LDAP_BASE_DN') ?: 'dc=example,dc=local',
    'bind_dn' => '%dn',
```

```
'bind_pass' => '%p',
'search_base_dn' => '',
'search_filter' => '(&(objectclass=inetOrgPerson)(uid=%u))',
'search_bind_dn' => getenv('LDAP_BIND_DN') ?: 'cn=admin,dc=example,dc=local',
'search_bind_pw' => getenv('LDAP_BIND_PASSWORD') ?: 'admin',
'scope' => 'list'
];
```

```
// === PRINCIPAL SEARCH ===
```

```
$config['principal']['ldap'] = [
    'base_dn' => getenv('LDAP_BASE_DN') ?: 'dc=example,dc=local',
    'search_base_dn' => '',
    'search_filter' => '(&(objectclass=inetOrgPerson)(uid=*))',
    'search_bind_dn' => getenv('LDAP_BIND_DN') ?: 'cn=admin,dc=example,dc=local',
    'search_bind_pw' => getenv('LDAP_BIND_PASSWORD') ?: 'admin',
    'scope' => 'list',
    'fieldmap' => [
        'id' => 'uid',
        'displayname' => 'cn',
        'mail' => 'mail'
    ]
];
```

```
// Hinweis: Die folgenden Einträge sind stark gekürzt. Siehe Original für volle Struktur.
```

```
// Du kannst z. B. $config['card']['addressbook']['ldap']['me'], ['global'], ['personal'] wie oben mit getenv() einbinden.
```

```
// Beispiel für ein Adressbuch-Eintrag mit bind_dn über ENV
```

```
$config['card']['addressbook']['ldap']['personal'] = [
    'name' => 'starface',
    'description' => 'Starface Kontakte',
    'user_specific' => true,
    'writable' => true,

    'group_LDAP_Object_Classes' => ['groupOfNames'],
    'group_required_fields' => ['cn', 'member'],
    'group_LDAP_rdn' => 'cn',
    'group_member_map' => [ 'MEMBER' => [ 'field_name' => 'member' ] ],

    'base_dn' => getenv('LDAP_BASE_DN') ?: 'dc=example,dc=local',
```

```

'filter' => '(objectClass=inetOrgPerson)',
'bind_dn' => getenv('LDAP_BIND_DN') ?: 'cn=admin,dc=example,dc=local',
'bind_pass' => getenv('LDAP_BIND_PASSWORD') ?: 'admin',
'scope' => 'sub',

'LDAP_Object_Classes' => ['inetOrgPerson'],
'required_fields' => ['cn','sn'],
'LDAP_rdn' => 'cn',

// Schreibrechte aktivieren
//"field_acl" => [
//  'eval' => 'w'
//  'list' => ['displayName', 'homePhone', 'telephoneNumber', 'facsimileTelephoneNumber', 'pager', 'mobile',
'homePostalAddress', 'preferredLanguage']
[] // leere Liste = alles erlaubt
// ],

// Vollständige Feldzuordnung für Outlook / CalDAV
'fieldmap' => [
  'FN' => ['field_name' => 'cn'],
  'N' => ['field_name' => [
    'last_name' => 'sn',
    'first_name' => 'givenName',
    'prefix' => 'personalTitle'
  ]],
  'EMAIL' => ['field_name' => 'mail'],
  'ORG' => ['field_name' => [
    'org_name' => 'o',
    'org_unit_name' => 'ou'
  ]],
  'TITLE' => ['field_name' => 'title'],
  'ROLE' => ['field_name' => 'employeeType'],
  'NICKNAME' => ['field_name' => 'displayName'],
  'PHOTO' => [[
    'field_name' => 'jpegphoto',
    'parameters' => [],
    'reverse_map_parameter_index' => 0,
    'decode_file' => true
  ]],
  'NOTE' => ['field_name' => 'description'],

```

```
'TEL' => [  
  [ // Fax number  
    'field_name' => 'facsimileTelephoneNumber',  
    'parameters' => [  
      ['TYPE' => ['fax']],  
      ['TYPE' => ['fax', 'work']],  
      ['TYPE' => ['work', 'fax']],  
      ['TYPE' => 'facsimile'],  
      ['TYPE' => ['voice', 'fax']],  
      ['TYPE' => ['fax', 'voice']],  
      null  
    ],  
    'reverse_map_parameter_index' => 0  
  ],  
  [ // Work number  
    'field_name' => 'telephoneNumber',  
    'parameters' => [  
      ['TYPE' => ['work']],  
      ['TYPE' => ['voice', 'work']],  
      ['TYPE' => 'work'],  
      ['TYPE' => 'voice'],  
      null  
    ],  
    'reverse_map_parameter_index' => 0  
  ],  
  [ // Home number  
    'field_name' => 'homePhone',  
    'parameters' => [  
      ['TYPE' => ['home']],  
      ['TYPE' => ['voice', 'home']],  
      ['TYPE' => 'home'],  
      null  
    ],  
    'reverse_map_parameter_index' => 0  
  ],  
  [ // Mobile number  
    'field_name' => 'mobile',  
    'parameters' => [  
      ['TYPE' => ['cell']],  
      ['TYPE' => ['voice', 'cell']],
```

```

        ['TYPE' => 'cell'],
        null
    ],
    'reverse_map_parameter_index' => 0
],
[ // Pager
    'field_name' => 'pager',
    'parameters' => [
        ['TYPE' => ['pager']],
        null
    ],
    'reverse_map_parameter_index' => 0
]
],
'NOTE' => [
    'field_name' => 'description',
    'parameters' => [],
    'reverse_map_parameter_index' => 0
],
'ADR' => [
    [
        'field_name' => [
            'po_box'    => 'postOfficeBox',
            'street'   => 'street',
            'locality' => 'l',
            'province' => 'st',
            'postal_code' => 'postalCode'
        ],
        'parameters' => ['TYPE' => 'work'],
        'map_component_separator' => ';',
        'reverse_map_parameter_index' => 0
    ],
    // Privatadresse
    [
        'field_name' => 'homePostalAddress',
        'parameters' => ['TYPE' => 'home'],
        'map_component_separator' => '$',
        'reverse_map_parameter_index' => 0
    ]
],
],

```

```
'LANG' => ['field_name' => 'preferredLanguage']
]
];
```

Version 2, nur schreibend mit einer Gruppe

```
<?php

$config = [];

// === TEMP / DATA ===
$config['tmpdir'] = '%systemtmpdir';
$config['datadir'] = '/var/www/html/ldap-carddav/data';

// === DATABASE ===
$config['sync_database'] = [
    'dsn' => 'sqlite:/var/www/html/ldap-carddav/data/cards.db',
    'username' => '',
    'password' => '',
    'options' => [],
    'init_commands' => []
];

// === LDAP SERVER CONFIG ===
$config['server']['ldap'] = [
    'host' => getenv('LDAP_HOST') ?: 'localhost',
    'network_timeout' => 10,
    'connection_security' => 'none'
];

// === LDAP AUTH ===
$config['auth']['ldap'] = [
    'base_dn' => getenv('LDAP_BASE_DN') ?: 'dc=example,dc=local',
    'bind_dn' => '%dn',
    'bind_pass' => '%p',
    'search_base_dn' => '',
    'search_filter' => '(&(objectclass=inetOrgPerson)(uid=%u))',
    'search_bind_dn' => getenv('LDAP_BIND_DN') ?: 'cn=admin,dc=example,dc=local',
    'search_bind_pw' => getenv('LDAP_BIND_PASSWORD') ?: 'admin',
    'scope' => 'list'
];
```

```

];

// === PRINCIPAL SEARCH ===
$config['principal']['ldap'] = [
    'base_dn' => getenv('LDAP_BASE_DN') ?: 'dc=example,dc=local',
    'search_base_dn' => '',
    'search_filter' => '(&(objectclass=inetOrgPerson)(uid=*))',
    'search_bind_dn' => getenv('LDAP_BIND_DN') ?: 'cn=admin,dc=example,dc=local',
    'search_bind_pw' => getenv('LDAP_BIND_PASSWORD') ?: 'admin',
    'scope' => 'list',
    'fieldmap' => [
        'id' => 'uid',
        'displayname' => 'cn',
        'mail' => 'mail'
    ]
];

// Funktion einfügen oder inkludieren
function getWritableUserIdsFromLdapGroup(): array {
    $groupCN = getenv('LDAP_WRITE_GROUP') ?: 'write';
    $ldapDomain = getenv('LDAP_DOMAIN') ?: 'example';
    $ldapTopDomain = getenv('LDAP_TOP_DOMAIN') ?: 'local';
    $groupDN = "cn={$groupCN},dc={$ldapDomain},dc={$ldapTopDomain}";

    $ldap = ldap_connect('localhost');
    ldap_set_option($ldap, LDAP_OPT_PROTOCOL_VERSION, 3);

    $bindDN = getenv('LDAP_BIND_DN') ?: "cn=admin,dc={$ldapDomain},dc={$ldapTopDomain}";
    $bindPW = getenv('LDAP_BIND_PASSWORD') ?: 'admin';

    if (!@ldap_bind($ldap, $bindDN, $bindPW)) {
        error_log("⚠ LDAP Bind fehlgeschlagen mit DN: $bindDN");
        return [];
    }

    $results = ldap_search($ldap, $groupDN, '(objectClass=groupOfNames)', ['member']);
    if (!$results) {
        error_log("⚠ Konnte Mitglieder von Gruppe $groupDN nicht abrufen.");
        return [];
    }
}

```

```

$entries = ldap_get_entries($ldap, $results);
if (!isset($entries[0]['member'])) {
    return [];
}

$uids = [];
foreach ($entries[0]['member'] as $key => $dn) {
    if (is_numeric($key) && preg_match('/uid=([^\,]+)/i', $dn, $matches)) {
        $uids[] = $matches[1];
    }
}

return $uids;
}

// Benutzer-ID abrufen
$user = $_SERVER['PHP_AUTH_USER'] ?? null;

// Schreibrechte prüfen
$writeUsers = getWritableUserIdsFromLdapGroup();
$isWritable = in_array($user, $writeUsers);

// Hinweis: Die folgenden Einträge sind stark gekürzt. Siehe Original für volle Struktur.
// Du kannst z. B. $config['card']['addressbook']['ldap']['me'], ['global'], ['personal'] wie oben mit getenv()
einbinden.

// Beispiel für ein Adressbuch-Eintrag mit bind_dn über ENV
$config['card']['addressbook']['ldap']['personal'] = [
    'name' => 'starface',
    'description' => 'Starface Kontakte',
    'user_specific' => true,
    'writable' => $isWritable,

    'group_LDAP_Object_Classes' => ['groupOfNames'],
    'group_required_fields' => ['cn', 'member'],
    'group_LDAP_rdn' => 'cn',
    'group_member_map' => [ 'MEMBER' => [ 'field_name' => 'member' ] ],

```

```

'base_dn' => getenv('LDAP_BASE_DN') ?: 'dc=example,dc=local',
'filter' => '(objectClass=inetOrgPerson)',
'bind_dn' => getenv('LDAP_BIND_DN') ?: 'cn=admin,dc=example,dc=local',
'bind_pass' => getenv('LDAP_BIND_PASSWORD') ?: 'admin',
'scope' => 'sub',

'LDAP_Object_Classes' => ['inetOrgPerson'],
'required_fields' => ['cn','sn'],
'LDAP_rdn' => 'cn',

// Schreibrechte aktivieren
//"field_acl" => [
//  'eval' => 'w'
//  'list' => ['displayName', 'homePhone', 'telephoneNumber', 'facsimileTelephoneNumber', 'pager', 'mobile',
'homePostalAddress', 'preferredLanguage']
[] [] [] [] [] // leere Liste = alles erlaubt
// ],

// Vollständige Feldzuordnung für Outlook / CalDAV
'fieldmap' => [
  'FN' => ['field_name' => 'cn'],
  'N' => ['field_name' => [
    'last_name' => 'sn',
    'first_name' => 'givenName',
    'prefix' => 'personalTitle'
  ]],
  'EMAIL' => ['field_name' => 'mail'],
  'ORG' => ['field_name' => [
    'org_name' => 'o',
    'org_unit_name' => 'ou'
  ]],
  'TITLE' => ['field_name' => 'title'],
  'ROLE' => ['field_name' => 'employeeType'],
  'NICKNAME' => ['field_name' => 'displayName'],
  'PHOTO' => [[
    'field_name' => 'jpegphoto',
    'parameters' => [],
    'reverse_map_parameter_index' => 0,
    'decode_file' => true
  ]],
],

```

```
'NOTE' => ['field_name' => 'description'],
'TEL' => [
  [ // Fax number
'field_name' => 'facsimileTelephoneNumber',
'parameters' => [
  ['TYPE' => ['fax']],
  ['TYPE' => ['fax', 'work']],
  ['TYPE' => ['work', 'fax']],
  ['TYPE' => 'facsimile'],
  ['TYPE' => ['voice', 'fax']],
  ['TYPE' => ['fax', 'voice']],
  null
],
'reverse_map_parameter_index' => 0
],
[ // Work number
'field_name' => 'telephoneNumber',
'parameters' => [
  ['TYPE' => ['work']],
  ['TYPE' => ['voice', 'work']],
  ['TYPE' => 'work'],
  ['TYPE' => 'voice'],
  null
],
'reverse_map_parameter_index' => 0
],
[ // Home number
'field_name' => 'homePhone',
'parameters' => [
  ['TYPE' => ['home']],
  ['TYPE' => ['voice', 'home']],
  ['TYPE' => 'home'],
  null
],
'reverse_map_parameter_index' => 0
],
[ // Mobile number
'field_name' => 'mobile',
'parameters' => [
  ['TYPE' => ['cell']],
```

```

    ['TYPE' => ['voice', 'cell']],
    ['TYPE' => 'cell'],
    null
  ],
  'reverse_map_parameter_index' => 0
],
[ // Pager
  'field_name' => 'pager',
  'parameters' => [
    ['TYPE' => ['pager']],
    null
  ],
  'reverse_map_parameter_index' => 0
]
],
'NOTE' => [
  'field_name' => 'description',
  'parameters' => [],
  'reverse_map_parameter_index' => 0
],
'ADR' => [
  [
    'field_name' => [
      'po_box'    => 'postOfficeBox',
      'street'   => 'street',
      'locality' => 'l',
      'province' => 'st',
      'postal_code' => 'postalCode'
    ],
    'parameters' => ['TYPE' => 'work'],
    'map_component_separator' => ';',
    'reverse_map_parameter_index' => 0
  ],
  // Privatadresse
  [
    'field_name' => 'homePostalAddress',
    'parameters' => ['TYPE' => 'home'],
    'map_component_separator' => '$',
    'reverse_map_parameter_index' => 0
  ]
]

```

```
],  
  'LANG' => ['field_name' => 'preferredLanguage']  
]  
];
```

Nun noch die. env erweitern. write ist hier der gruppenname

```
...  
LDAP_WRITE_GROUP=write  
...
```

Nun noch wieder das Adressbuch mit syncdb.php löschen neu anlegen, siehe [hier](#)

Aufrufen LDAP und co:

- **phpLDAPadmin:** <https://localhost:6443>
Benutzername aus unserem Beispiel : cn=admin,dc=example,dc=local
Passwort aus unserem Beispiel : admin



Authenticate to server ldap

Login DN:
cn=admin,dc=example,dc=local

Password:
.....

Anonymous

Authenticate

- **ldap-carddav WebDAV/CardDAV:** <http://localhost/ldap-carddav/>
Die Benutzer dazu werden im LDAP Webgui angelegt, dazu ein ein eigenes Kapitel

Datenbank initialisieren

```
docker-compose exec carddav /bin/bash  
sqlite3 /var/www/html/ldap-carddav/data/cards.db < /var/www/html/ldap-carddav/sql/sqlite/ddl.sql  
php /var/www/html/ldap-carddav/src/App/syncdb.php init
```

Danach vom container wieder abmelden und chmod 777 über die card.db

```
chown www-data:www-data -R /var/www/html/ldap-carddav/data
```

PHP ini Änderungen durchführen und neu mit der Datenbank synchronisieren:

Wenn die PHP geändert wird um zum Beispiel Felder hinzugefügt werden, muss die Datenbank neu initialisiert werden.

```
docker-compose exec carddav /bin/bash  
php /var/www/html/ldap-carddav/src/App/syncdb.php
```

Ausgabe:

Dort 0 Auswählen

```
Choose the entity you want to operate upon. Enter 0 for addressbook and 1 for user:
```

Nun mit 3 bestätigen

```
Enter the operation to perform on address book. Enter 0 to list, 1 to add, 2 to rename and 3 to delete:
```

Nun Adressbuchname eingeben:personal.

```
Enter name of the address book to delete:
```

Nun wurde das Buch gelöscht

```
Address book 'personal' has been deleted.
```

Nun kann ein neuer init stattfinden

```
php /var/www/html/ldap-carddav/src/App/syncdb.php init
```

Nun die Rechte neu setzen

```
chown www-data:www-data -R /var/www/html/ldap-carddav/data
```

Ausgabe:

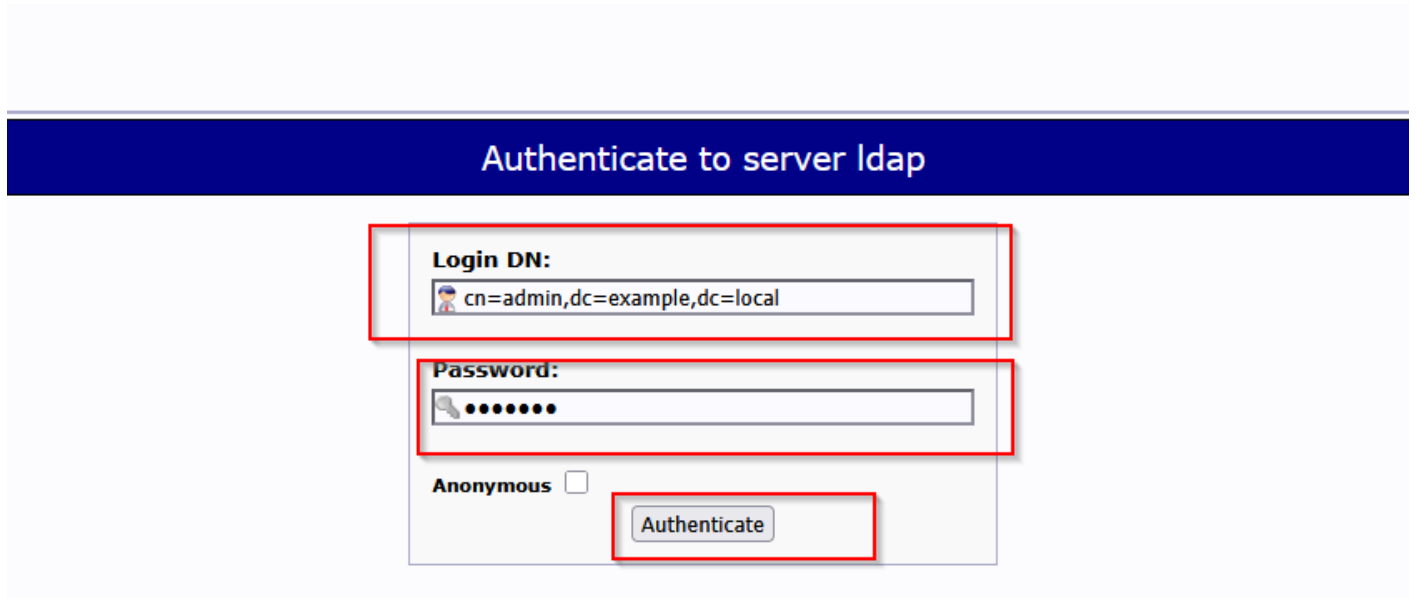
```
Initializing sync database ...  
Address book 'personal' has been successfully added to sync database.
```

Address book(s) successfully imported.

Benutzer anlegen:

Im LDAP Webgui einloggen unter

https://<ip>:6443

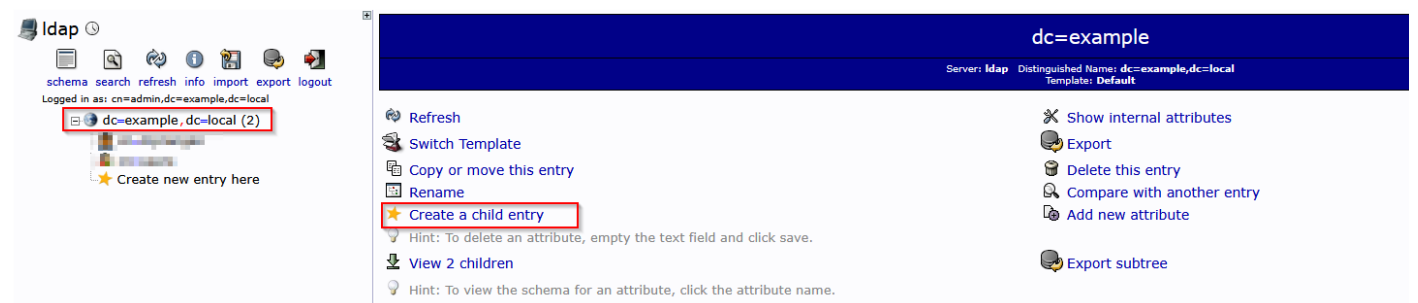


The image shows a web interface for authenticating to an LDAP server. At the top, there is a dark blue header with the text "Authenticate to server ldap" in white. Below the header is a light gray form area. The form contains three main sections, each highlighted with a red rectangular box:

- Login DN:** A text input field containing the string "cn=admin,dc=example,dc=local".
- Password:** A password input field with a masked password represented by ten black dots.
- Anonymous:** A checkbox that is currently unchecked.

At the bottom right of the form area is a button labeled "Authenticate", which is also highlighted with a red rectangular box.

Dort neues child Element anlegen...



The image displays the LDAP web GUI interface. On the left side, there is a navigation pane with a tree view showing the directory structure. The entry "dc=example, dc=local (2)" is highlighted with a red rectangular box. Below the tree view, there is a "Create new entry here" link.

The main content area is titled "dc=example" and shows the details of the selected entry. The "Server" is "ldap" and the "Distinguished Name" is "dc=example,dc=local". The "Template" is "Default".

On the left side of the main content area, there is a list of actions:

- Refresh
- Switch Template
- Copy or move this entry
- Rename
- Create a child entry** (highlighted with a red rectangular box)
- View 2 children

On the right side of the main content area, there is a list of actions:

- Show internal attributes
- Export
- Delete this entry
- Compare with another entry
- Add new attribute
- Export subtree

At the bottom of the main content area, there are two hints:

- Hint: To delete an attribute, empty the text field and click save.
- Hint: To view the schema for an attribute, click the attribute name.

Vom Typ PosixGroup

Templates:

- Courier Mail: Account
- Courier Mail: Alias
- Generic: Address Book Entry
- Generic: DNS Entry
- Generic: LDAP Alias
- Generic: Organisational Role
- Generic: Organisational Unit
- Generic: Posix Group**
- Generic: Simple Security Object
- Generic: User Account
- Kolab: User Entry
- Samba: Account
- Samba: Domain
- Samba: Group Mapping
- Samba: Machine
- Sendmail: Alias
- Sendmail: Cluster
- Sendmail: Domain
- Sendmail: Relays
- Sendmail: Virtual Domain
- Sendmail: Virtual Users
- Thunderbird: Address Book Entry
- Default

mit dem namen users

Create Object

Server: **ldap** Container: **dc=example,dc=local**
Template: **Generic: Posix Group (posixGroup)**

New Posix Group (Step 1 of 1)

Group alias, required, rdn

GID Number alias, required, hint, ro

Users alias, hint

Nun nochmals bestätigen

Do you want to create this entry?









Attribute	New Value	Skip
cn=users,dc=example,dc=local		
Group	users	<input type="checkbox"/>
GID Number	500	<input type="checkbox"/>
objectClass	posixGroup	<input type="checkbox"/>

nun eine weitere elemnt vom typ organizationRole mit den namen write anlegen

Create Object

Server: **Idap** Container: **dc=example,dc=local**

Select a template for the creation process

-  Courier Mail: Account
-  Courier Mail: Alias
-  Generic: Address Book Entry
-  Generic: DNS Entry
-  Generic: LDAP Alias
-  **Generic: Organisational Role**
-  Generic: Organisational Unit
-  Generic: Posix Group
-  Generic: Simple Security Object
-  Generic: User Account
-  Kolab: User Entry
-  Samba: Account
-  Samba: Domain
-  Samba: Group Mapping
-  Samba: Machine
-  ~~Sendmail: Alias~~
-  ~~Sendmail: Cluster~~
-  ~~Sendmail: Domain~~
-  ~~Sendmail: Relays~~
-  ~~Sendmail: Virtual Domain~~
-  ~~Sendmail: Virtual Users~~
-  Thunderbird: Address Book Entry
-  Default

nun den Namen vergeben

Create Object

Server: **Idap** Container: **dc=example,dc=local**
Template: **Generic: Organisational Role (organizationalRole)**

New Organisational Role (Step 1 of 1)

Role CN

alias, required, rdn

write *

runter scrollen bis create object

State alias

Postal code alias

Postal Address alias

Registered Address alias

Nun einen Benutzer anlegen vom typ inetOrgPerson

<input type="radio"/> Courier Mail: Account <input type="radio"/> Courier Mail: Alias <input type="radio"/> Generic: Address Book Entry <input type="radio"/> Generic: DNS Entry <input type="radio"/> Generic: LDAP Alias <input type="radio"/> Generic: Organisational Role <input type="radio"/> Generic: Organisational Unit <input type="radio"/> Generic: Posix Group <input type="radio"/> Generic: Simple Security Object <input type="radio"/> Generic: User Account <input type="radio"/> Kolab: User Entry <input type="radio"/> Samba: Account	<input type="radio"/> Samba: Domain <input type="radio"/> Samba: Group Mapping <input type="radio"/> Samba: Machine <input checked="" type="radio"/> Sendmail: Alias <input checked="" type="radio"/> Sendmail: Cluster <input checked="" type="radio"/> Sendmail: Domain <input checked="" type="radio"/> Sendmail: Relays <input checked="" type="radio"/> Sendmail: Virtual Domain <input checked="" type="radio"/> Sendmail: Virtual Users <input type="radio"/> Thunderbird: Address Book Entry <input type="radio"/> Default
--	--

Daten ausfüllen

New User Account (Step 1 of 1)

Common Name ist der Anzeigename im Baum

First name alias

 Max

Last name alias, required

Mustermann

Common Name alias, required, rdn

Max Mustermann

User ID alias, required

mmustermann

Das ist der Benutzername, dieser heißt nach dem speichern User Name

Password alias, hint



••••

md5

••••

(confirm)

[Check password...](#)

UID Number alias, required, hint, ro

 1000

GID Number alias, required, hint

users

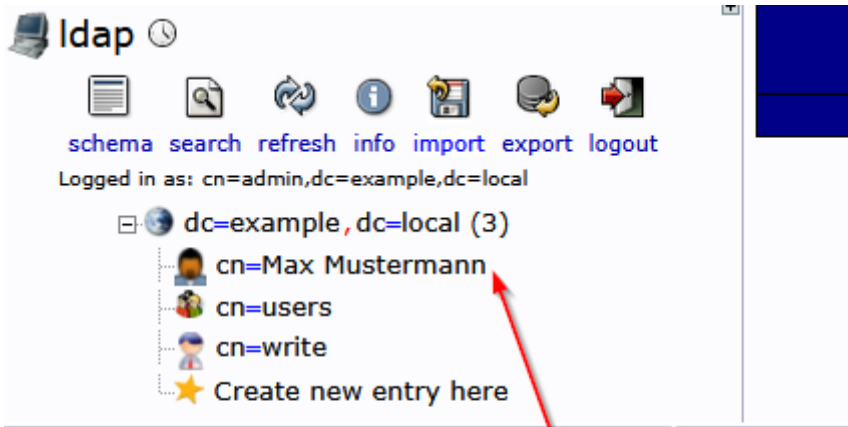
Home directory alias, required

/home/users/mmustermann

Login shell alias

Create Object

Nun sieht das ganz so aus:



Common name
(Anzeigename)

Nun den benutzer im carddav Backend anlegen, nicht im Idap, das haben wir gerade getan.

```
docker-compose exec carddav /bin/bash  
php /var/www/html/ldap-carddav/src/App/syncdb.php
```

Fehlersuche:

Invalid Credentials:

Sollte Verbindung nicht zustande kommen, dann mal in die Apache2 log schauen.

Im container anmelden

```
docker-compose exec carddav /bin/bash  
cat /var/log/apache2/error.log
```

Ausgabe:

```
[Sun May 18 09:28:57.236797 2025] [php:warn] [pid 13:tid 13] [client 172.0.0.2:61975] PHP Warning: ldap_bind(): Unable to bind to server: Invalid credentials in /var/www/html/ldap-carddav/src/DAV/Utility/LDAP.php on line 87  
[Sun May 18 09:28:57.237857 2025] [php:notice] [pid 13:tid 13] [client 172.0.0.2:61975] Could not establish bind connection to backend server in ISubsoft\\DAV\\Auth\\Backend\\LDAP::validateUserPass at line 65  
[Sun May 18 09:29:15.836070 2025] [php:warn] [pid 14:tid 14] [client 172.0.0.2:62681] PHP Warning: ldap_bind(): Unable to bind to server: Invalid credentials in /var/www/html/ldap-carddav/src/DAV/Utility/LDAP.php on line 87  
[Sun May 18 09:29:15.836147 2025] [php:notice] [pid 14:tid 14] [client 172.0.0.2:62681] Could not establish bind connection to backend server in ISubsoft\\DAV\\Auth\\Backend\\LDAP::validateUserPass at line 65  
[Sun May 18 09:31:28.821090 2025] [php:warn] [pid 9:tid 9] [client 172.0.0.2:62874] PHP Warning: ldap_bind(): Unable to bind to server: Invalid credentials in /var/www/html/ldap-carddav/src/DAV/Utility/LDAP.php on line 87  
[Sun May 18 09:31:28.821189 2025] [php:notice] [pid 9:tid 9] [client 172.0.0.2:62874] Could not establish bind connection to backend server in ISubsoft\\DAV\\Auth\\Backend\\LDAP::validateUserPass at line 65  
root@7dcf55307843:/var/www/html/ldap-carddav#
```

Testen der Daten

Hinweis: Paramter -w ist das Passwort aus der .env Datei für ldap

```
ldapwhoami -x -D "cn=admin,dc=example,dc=local" -w admin -H ldap://ldap
```

Ausgabe:

```
root@44a0bc55808:/var/www/html/ldap-carddav# ldapwhoami -x -D "cn=admin,dc=example,dc=local" -w admin -H ldap://ldap
dn:cn=admin,dc=example,dc=local
```

Testen ob ein Objekt sich ändern lässt

```
ldapmodify -x -D "cn=admin,dc=example,dc=local" -w admin -H ldap://ldap
```

Einfügen und dann strg+d drücken

```
dn: cn=shacker,dc=example,dc=local
changetype: modify
replace: mail
mail: test@example.org
```

Ein komplettes Objekt ausgeben

```
ldapsearch -x \
-D "cn=admin,dc=example,dc=local" \
-w admin \
-b "cn=Wolf, SDaniel,dc=example,dc=local" \
-LLL
```

Ausgabe:

```
dn: cn=Wolf\2C SDaniel,dc=example,dc=local
objectClass: inetOrgPerson
cn: Wolf, SDaniel
sn: Wolf
givenName: SDaniel
mail: daniel.wolf@test.de
```

Database readonly:

Fehler aus der /var/log/apache2/error.log

```
attempt to write a readonly database, referer: http://192.168.0.231/server.php/addressbooks/mprangen/
[Thu Jul 31 19:11:24.152676 2025] [php7:notice] [pid 9:tid 9] [client 192.168.0.26:51450] Database query could
not be executed: ISubsoft\\DAV\\CardDAV\\Backend\\LDAP::fullSyncOperation at line no 1856, SQLSTATE[HY000]:
General error: 8 attempt to write a readonly database, referer:
http://192.168.0.231/server.php/addressbooks/mprangen/
[Thu Jul 31 19:12:16.335653 2025] [php7:notice] [pid 12:tid 12] [client 192.168.0.26:51452] Database query
could not be executed: ISubsoft\\DAV\\CardDAV\\Backend\\LDAP::fullSyncOperation at line no 1856,
SQLSTATE[HY000]: General error: 8 attempt to write a readonly database, referer:
http://192.168.0.231/server.php/addressbooks/mprangen/
[Thu Jul 31 19:12:18.828919 2025] [php7:notice] [pid 13:tid 13] [client 192.168.0.26:51453] Database query
could not be executed: ISubsoft\\DAV\\CardDAV\\Backend\\LDAP::fullSyncOperation at line no 1856,
SQLSTATE[HY000]: General error: 8 attempt to write a readonly database, referer:
http://192.168.0.231/server.php/addressbooks/mprangen/
```

Wenn syncdb.php ohne Probleme ausgeführt werden kann, liegt es nicht an Dateirechten bei root, sondern bei www-datat user.

wir werden das Verzeichnis nochmal neu mit rechten vergeben.

In den Container einloggen

```
docker-compose exec carddav /bin/bash
```

Dnn rechte vergeben

```
chown www-data:www-data -R /var/www/html/ldap-carddav/data
```

Ansonsten schauen wir uns mal an mit welchen umser der apache2 ausgeführt wird

```
ps aux | grep apache

root      1  0.0  0.0  2480  580 ?      Ss   19:32   0:00 /bin/sh /usr/sbin/apachectl -D FOREGROUND
root      8  0.0  1.4 206336 28308 ?      S    19:32   0:00 /usr/sbin/apache2 -D FOREGROUND
www-data  9  0.0  1.1 208744 23500 ?      S    19:32   0:00 /usr/sbin/apache2 -D FOREGROUND
www-data 10  0.0  1.1 208876 24056 ?      S    19:32   0:00 /usr/sbin/apache2 -D FOREGROUND
www-data 11  0.0  1.1 208876 23536 ?      S    19:32   0:00 /usr/sbin/apache2 -D FOREGROUND
www-data 12  0.0  1.1 208876 23556 ?      S    19:32   0:00 /usr/sbin/apache2 -D FOREGROUND
www-data 13  0.0  0.4 206376  9264 ?      S    19:32   0:00 /usr/sbin/apache2 -D FOREGROUND
root     42  0.0  0.0  3240  648 pts/0  S+   19:38   0:00 grep apache
```

Version #48

Erstellt: 17 Mai 2025 23:44:20 von Admin

Zuletzt aktualisiert: 1 August 2025 06:42:12 von Admin