



labfolder Installation Manual

v1.24.2

Table of Contents

Table of Contents	2
Requirements for local installation	4
Hardware and software requirements	4
labfolder Client Requirements	4
General infrastructure of the labfolder application	4
Backup and recovery	5
Long-term archiving of labfolder data	6
High availability, redundancy and failover	6
Update of the operating system and components	7
Capacities and Scaling	7
Migrating labfolder	8
External connections	8
Installing and updating labfolder	8
1. Install mysql	8
2. Install Docker	9
3. Install labfolder	9
4. Configure labfolder	10
Setting the default domain	10
Connecting your MySQL database	10
Set your timezone	10
Connecting your mail account	11
Configuring support	12
Customize system mails	12
Configure server event logging	12
Setting the maximum upload file size	13
Configure default user and group settings	13
Global control of data safeguarding	14
Terms of use and privacy statement	15
LDAP Authentication	15
Activate LDAP	15
Authenticate with DN	16
Authenticate with Attribute	16

Update existing users after LDAP activation	16
Examples	17
Active Directory	17
Configuring usage-reporting	18
5. Create the database	19
6. Run labfolder	19
7. Update labfolder	19
Activating Apps	20
XHTML	20
Mendeley	21
Dropbox	21
figshare	22
Autostart labfolder	22

Requirements for local installation

Hardware and software requirements

For the local installation of labfolder, the following requirements have to be met:

Hardware:

- Min. 8 GB RAM
- Hard drive min. 500 GB (higher data volumes require the installation of more hard drive space)

Software :



Linux Operating System that is compatible with the latest version of Docker CE (min. 17.12.0-ce), we recommend a recent version Ubuntu OS. A list of supported Linux Operating Systems can be found here: <https://docs.docker.com/engine/installation/>

- Docker CE (min. version 17.12.0-ce) for running the labfolder application package on physical or virtual servers
- MySQL database, min. version 5.7
- Mysql Tools (mysql, mysqldump)
- Connection to e-mail server with SMTP protocol and an e-mail account for the sending of notifications to users (optional)
- static IP address
- for remote installation, updates and maintenance: remote access (via SSH, root privileges not required)

Updates for the labfolder software will be distributed for self-installation, with the possibility to request remote assistance for the installation. Software updates of the operating system and external software components are within the responsibility of the customer.

labfolder Client Requirements

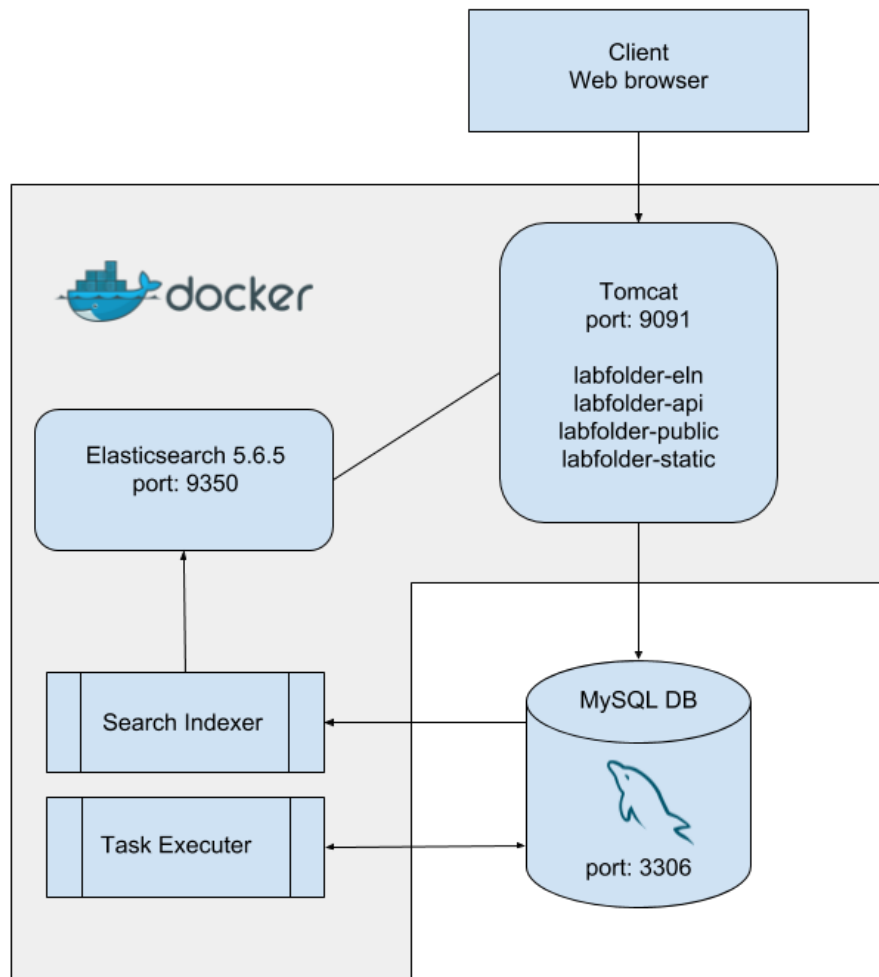
The labfolder software and user interface is accessible from any modern web browser (independent of the operating system). The supported browsers are listed on our website: <https://www.labfolder.com/browser-support/>

General infrastructure of the labfolder application

Generally, the labfolder application consist of two parts: A Docker container containing all components of the labfolder application and a MySQL database.

This architecture and the underlying licensing terms make it necessary to pre-install Docker and to either install a MySQL database on the same server or give the labfolder application access to a database server.

The high-level architecture of labfolder and its main components is summarized in the graph below:



Backup and recovery

For the backup and recovery of labfolder, mysqldump can be used. For a detailed overview on how to set-up mysqldump for backup and recovery, please refer to the official documentation at <http://dev.mysql.com/doc/refman/5.7/en/mysqldump.html>. Please note that the contractual liability of labfolder in case of data loss is limited to the data loss that would have occurred in case of updates being made on a weekly basis. We do, however, recommend to make updates on a daily basis.

Long-term archiving of labfolder data

Generic hard drives, as they are being used in the most common commercial server infrastructures, usually have a life cycle of several years, depending on the manufacturer. In order to prevent data loss in the event of hardware failure, it is thus recommended to provide daily updates and consider a regular dump of the labfolder database to a long-term storage system.

We recommend the installation of a redundant storage hardware (i.e. RAID) and to implement regular backups of the labfolder database. In such a setup, defective hard drives can be replaced without causing any data loss. Additionally, this set-up allows the secure storage of data beyond the life cycle of a singular hard drive.

The labfolder installation can also be connected to a database server, through which updates can be managed as well.

While in such a setup, data can be kept in labfolder for long-term archiving, it is desirable that for long-term storage, specific data can be moved to a long term storage archive in order to provide an economic storage of larger data quantities and to avoid an overflow of the labfolder database.

For the extraction of labfolder data from the database and the subsequent long-term archiving, the following procedure can be applied:

1. Scientists or Principal Investigators which are responsible for the management and the archiving of scientific data can extract the relevant data via labfolder's XHTML export. This export allows the extraction of specific data which belong to specific responsible persons or projects.
2. After export, scientists or principal investigators receive a .zip archive which contains all the exported lab notebook entries in a html format which provides an offline read-only version of the exported data, together with a folder structure containing all relevant files and thus provides a complete export of all relevant data.
3. The exported .zip archive can be used for long-term archiving on a dedicated long term archiving system.
4. After archiving, scientists or principal investigators can delete the data from the labfolder database.

Details on the XHTML export can be found in the labfolder user manual, which is available on our website at <https://www.labfolder.com/manuals-tutorials/>

High availability, redundancy and failover

High availability and failover of the labfolder application can be established both on the database level and on the application level.

For high availability and failover on the database level, we recommend using database replication, which also helps to prevent data loss in case of database failure: <https://dev.mysql.com/doc/refman/5.7/en/replication.html>.

Replicated databases can be switched during failover to provide a continuous service of the database functionalities: <https://dev.mysql.com/doc/refman/5.7/en/replication.html>

On the application level, a secondary (virtual) server can be cloned by setting up a copy of the docker container in the same configuration to instantaneously replace the primary application in case of failure.

Note that for the simultaneous running of two labfolder application servers with a link to the same database (i.e. managed by a load balancer which can, for example, be provided by *nginx*), asynchronous behaviour may be observed at the level of the Search Indexer and the Task Executor, which may result in asynchronous search behaviour and double execution of notifications and tasks.

Update of the operating system and components

If an update of the operating system, required components like Docker and the MySQL database are required, we recommend stopping the application as recommended for an application update and explained in detail in the following installation instructions. After the update, the labfolder installation needs to be restarted as described in the instructions.

In case of a database update, we recommend creating a backup of the database before the update.

Without the time required for the database update, an update of labfolder or external components usually requires only a few minutes of downtime. User sessions are not terminated during restart, as they are stored in the database, but user activities like i.e. saving might be interrupted during restart. We still recommend informing users well in advance and calculate a generous window of service downtime.

Capacities and Scaling

On a medium size server as described in the specifications above, the labfolder applications handles several hundred simultaneous users without effort. For a larger number of simultaneous users, we recommend allowing for more extensive hardware requirements.

On average, an active group of 15 scientists generates below 10 GB of Data per year. This number can increase if automated methods or the API are used for regular bulk upload of data. User quota and maximum upload file size can be controlled by the administrator.

Should the database size become an issue, sharding of the database and setting up parallel installations is possible as well, since the labfolder license model does not restrict the number of labfolder server instances you are running.

Migrating labfolder

When migrating a labfolder database from one application or another, always make sure that

- Both labfolder versions updated to the same version
- Both MySQL versions are updated to the same version
- Both labfolder applications with a connected database have been initialized at least once (with the command `$./labfolder.sh start`)

External connections

If not configured otherwise, the labfolder application might attempt to make connections to the following external services:

Google - if `ACTIVE_USER_REPORT_USE_DEFAULT_MAIL_CLIENT=false`, labfolder will attempt to connect to gmail in order to send usage reports to labfolder via port 25. If your contractual agreement requires the sending of regular usage reports, make sure that either port 25 is open for the labfolder application or you use your own mail account for sending user reports by setting `ACTIVE_USER_REPORT_USE_DEFAULT_MAIL_CLIENT=true`.

Amazon S3 - if `EXPORT_DOCUMENT_REPOSITORY_TYPE=amazonS3`, labfolder will attempt to connect to Amazon S3 for the storage of temporary XHTML Export files. In order to store export files locally, set `EXPORT_DOCUMENT_REPOSITORY_TYPE=fileSystem` which will prevent connections to Amazon S3.

Cloudflare - labfolder will attempt a connection to cloudflare for the accelerated pre-loading of web fonts and web scripts. Currently, these resources are not available offline, thus, a connection to cloudflare is necessary to pre-load external resources and allow a flawless display of the user interface. No personal or user data is sent to cloudflare at any time.

Installing and updating labfolder

1. Install mysql

Download and install mysql-server on the same or on a different server you want labfolder to run from <http://dev.mysql.com/downloads/mysql/>. In case you do not want labfolder to use the root account, you have to create an account for labfolder. In case labfolder will be installed on a different server, you have to make sure to enable remote access to mysql.

If you would like to store all information in the database on a different drive or partition, you need to set the `datadir` of the mysql-server accordingly during installation following the instructions given in the documentation of mysql here:

<https://dev.mysql.com/doc/refman/5.7/en/data-directory-initialization-mysqld.html>

If you would like to change the data directory (i.e. the location, drive or partition where labfolder is stored) after the installation, instructions vary depending on the operating system used. For the most popular operating systems used for labfolder servers, instructions can be found as follows:



Ubuntu:

<https://www.digitalocean.com/community/tutorials/how-to-move-a-mysql-data-directory-to-a-new-location-on-ubuntu-16-04>



CentOS:

<https://www.digitalocean.com/community/tutorials/how-to-change-a-mysql-data-directory-to-a-new-location-on-centos-7>



Redhat:

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Managing_Confined_Services/sect-Managing_Confined_Services-MySQL-Configuration_Examples.html

2. Install Docker

1. Install Docker CE according to these instructions:



Ubuntu: <https://docs.docker.com/install/linux/docker-ce/ubuntu/>



Debian: <https://docs.docker.com/install/linux/docker-ce/debian/>



CentOS: <https://docs.docker.com/install/linux/docker-ce/centos/>



Fedora: <https://docs.docker.com/install/linux/docker-ce/fedora/>



On Linux, run `$ sudo usermod -aG docker <user>`
where `<user>` is the user you logged in with.

2. Reboot your system.

3. Install labfolder

1. Download labfolder.zip from the download link you received together with your license and put it in the directory in which you would like your labfolder installation files to be in. You can use the command `$ wget "download link"` as well.
2. Extract labfolder.zip. You might need to install unzip first.
3. You can verify your labfolder installation by running the command

 \$./labfolder.sh verify

The script will return OK if the hash sum of your download has been verified.

4. Configure labfolder

Configurations in labfolder can be made by editing the *server.cnf* file in the labfolder directory.

Setting the default domain

The default domain needs to be set for correct link generation in mails (for instance password recovery) in the section `#Network settings`.

1. Set `DEFAULT_DOMAIN` to the DNS or IP address of your server.
2. Set `DEFAULT_HTTP_PROTOCOL=` either to `https://` or to `http://` depending on the protocol you are using.

Please note that this configuration only changes the placeholder in system links and messages labfolder sends and does not configure the actual protocol of the labfolder server. The correct setting of this parameter is necessary for password recovery and data exports to work properly.

Connecting your MySQL database

Parameters for connecting your MySQL database are found in the section `#JDBC Properties` in the *server.cnf* file.



Linux: If your mysql is not installed on the same server as labfolder, replace `localhost` in `JDBC_DATABASE_URL` with the IP or DNS of your MySQL server. If you would like to establish a secure connection to your MySQL server, please follow the guidelines listed in the standard documentation of MySQL: <https://dev.mysql.com/doc/refman/5.5/en/secure-connections.html>

- Change `JDBC_USERNAME` to the mysql-username you want to use.
- Change `JDBC_PASSWORD` to the password of your username

Set your timezone

In order for your labfolder server to display correct server times in the ELN and to allow for the selection of time zones relative to the server time within the application, the timezone of your server needs to be set in the section `#JDBC Properties` in the *server.cnf* file.

Set the variable `JDBC_SERVER_TIMEZONE` to the timezone that you would like to set for your server. The value set for `JDBC_SERVER_TIMEZONE` must follow the nomenclature as listed in the column `TZ` in the list of `tz` database timezones:

https://en.wikipedia.org/wiki/List_of_tz_database_time_zones#List


To see which timezone your MySQL database is currently using, run the command

```
cat /etc/timezone
```

on your server.

Connecting your mail account

To connect a mail account to the labfolder application for password retrieval, messaging, sending of support tickets to labfolder and usage reporting, change the parameters in the `#Mail client properties` section as described below.

 Please note that labfolder uses default parameters for all variables and will attempt to connect to the labfolder mail servers if variables are not set in the `#Mail client properties` section. Thus, you need to remove the `#` symbol from all variables in this section if you want to connect your own account and set the parameters accordingly or leave them blank.

1. Remove the `#` symbol in front of all lines in the paragraph when connecting a mail account
2. Set the parameter `MAIL_STARTTLS=true` if you are using `STARTTLS` as authentication encryption, otherwise, comment it out by placing a `#` symbol in front of the line
3. Set the parameter `MAIL_AUTHENTICATION_ENABLE=true` if your mail server requires authentication, otherwise, comment it out by placing a `#` symbol in front of the line
4. Enter the URL of your mail server behind the `MAIL_HOST=` parameter
5. Set the `MAIL_PORT=` parameter to the active port of your mail server, default is `25` for `STARTTLS`
6. Set the `MAIL_USERNAME=` parameter to the username required for login, if no login is required, comment it out by placing a `#` symbol in front of the line
7. Set the `MAIL_PASSWORD=` parameter to the username required for login, if no login is required, comment it out by placing a `#` symbol in front of the line
8. Set the `MAIL_EMAIL=` to the email address your connected account is using.

Please note that the connected mail account needs to have the permission to pass the firewall if support requests and usage reports shall be sent to the labfolder team. Usage reports can be disabled separately.

Configuring support

When end users trigger a support request from the labfolder tool using the 'Support' button, an email is sent to the labfolder support team using the email account which is connected as described in the previous step. The receiving mail address is feedback@labfolder.com by default.

With the variable `#Mail Support Button` in the `server.cnf` file, you can set an alternative email address for the reception of first line support ticket by assigning the target email address to the variable `MAIL_EMAIL_FEEDBACK=`.

When leaving the variables commented out with a leading `#` symbol, the default address feedback@labfolder.com is used as a recipient for support requests and end users will receive support from the labfolder Support Team.

Please feel free to send a test support request to labfolder support team using the 'Support' Button for testing your connection.

Customize system mails

The labfolder system uses a repository of mail templates for sending mails after registration, after being invited to a group, for password retrieval and more. These mail templates are located in the folder `Labfolder/files/mail`.

For each mail template, there are up to three files:

1. A `.html` file containing the HTML version of a mail
2. A `.txt` file containing the text version of the mail, which is used as fallback if HTML is disabled
3. A `.subject` file containing the email subject in text format.

Please note that for a editing of an email template, edits have to be made to all files to be reflected in both subject and HTML (if available) and text versions of a file.

Placeholders in curly brackets (i.e. `{0}`, `{1}` etc.) are template-specific placeholders for variables like mail address, name, url etc. We recommend that you leave these placeholders unchanged when editing email templates.

- ⚠ Please note that when updating an existing installation, you will need to copy the files in the `labfolder/files/mail` folder from the update package to the same folder in your labfolder directory.

Configure server event logging

In the section `#Server Event logging` section, set `LOG_TO_FILE=true` to enable logging of server events to the `Labfolder/Files/*` folder. We recommend enabling event logging for a more detailed error reporting.

Setting the maximum upload file size

To configure the maximum size users can upload to the labfolder database, in the `#Maximum upload file size` section, set the `FILEUPLOAD_MAXUPLOADSIZE=` to the according size in bytes (default is 25000000).

! Please note that when configuring a proxy via *nginx*, the maximum file size for upload defaults to 2 MB. It might be necessary to adjust the maximum file size in *nginx* following the instructions on this website: <http://stackoverflow.com/questions/26717013/how-to-edit-nginx-conf-to-increase-file-size-upload>

! Please note that the [max_allowed_packet](https://stackoverflow.com/questions/8062496/how-to-change-max-allowed-packet-size) size of your MySQL database might further limit the maximum upload file size. You can adjust the `max_allowed_packet` size via editing the `my.cnf` file: <https://stackoverflow.com/questions/8062496/how-to-change-max-allowed-packet-size>

Configure default user and group settings

To control user and group settings, edit the `#User and group control` section as follows:

1. Set `DEFAULT_GROUP_SIZE` to the maximal number of seats a group should have when it is created. The fallback in case of a missing configuration variable is 3.
2. Set `DEFAULT_GROUP_TYPE_MAXI=true` if a new group should be created as maxi group. MAXI groups have advanced sharing and admin options.
3. Set `DEFAULT_USER_STORAGE` to the number of bytes that a newly created user should have to store his data. The default is 3 GB. Users will be unable to upload files when the quota set in this variable has been reached and will get an error warning. Users will still be able to add text and other content.

! Please note that the `DEFAULT_GROUP_SIZE` needs to be increased and the `DEFAULT_GROUP_TYPE_MAXI` needs to be set to `true` if large groups with advanced sharing and admin options should be available.

If `DEFAULT_GROUP_SIZE` is set to a value < 1 , the creation of new groups is not possible. This can be used to prevent the creation of groups by anyone on the server and i.e. only allow Principal Investigators the creation of groups during server set-up and in controlled intervals. As an IT administrator, you can also create groups yourself and 'transfer' them to Principal investigators later. For details, refer to the labfolder admin manual.

The default user and group settings can be used to control the size and mode of newly created groups on a running system, the settings of existing groups remain unchanged.

To change the group size and group type of all groups which are already existing, follow these steps:

1. Log on to the MySQL database via `mysql -u<user> -p<password>`, followed by `use labfolder`
2. Execute the SQL statement `UPDATE 'group' SET 'type' = '2'` to set the group type of the existing group to MAXI.
3. Execute the SQL statement `UPDATE 'group' SET 'seats' = '<group size>'` to set the seat number of the existing groups to `<group size>`

Global control of data safeguarding

Content can be hidden from entries, and projects and templates can be either hidden or terminally deleted. The terminal deletion of content in labfolder groups can be prevented by group admins (see also the section 'Data safeguarding' in the [labfolder admin manual](#)).

Data safeguarding preventing the terminal deletion of data can be globally activated for the entire server by setting the variable in the section `#User and group control`:

When `FEATURE_GLOBAL_PREVENT_DELETE_CONTENT=true`, data safeguarding is globally activated and cannot be deactivated by group admins. Additionally, groups and private projects cannot be deleted when global data safeguarding is activated. The default value is false, with global safeguarding being deactivated.

Upon global activation of data safeguarding, user data is not deleted when users close their accounts. Instead, the account is deactivated and no longer accessible to the user, but the data can be retrieved by the administrator any time with the following steps:


1. Connect to the mysql server and authenticate.
2. Run the following command:

```
USE <database-name>;  
  
UPDATE `user` SET  
`email`='<email-address-you-can-access>', `status`=0 WHERE  
`email` LIKE '||%||<email-address-of-deleted-account>';
```

Where `<email-address-you-can-access>` is an email address you have access to and `<email-address-of-deleted-account>` is the email address with which the deleted account is registered.

3. Go to the notebook login page and follow the "reset password" link to reset the password for the inserted email address `<email-address-you-can-access>` specified in the previous step.
4. You will get a email providing a link to assign a new password.

5. After logging in to the account with the new credentials, you can change the email address in the settings. Please make sure to store the credentials especially when changing the email to an invalid address.

 Please note that restoring user data as described above is only possible when global data safeguarding is activated.

Please keep in mind that when activating global data safeguarding, data cannot be deleted from the database, which will lead to a continuing increase in database size.

Terms of use and privacy statement


On the registration page of labfolder, URLs for the terms of use which have to be accepted can be customized in the section `#Terms and Privacy links`:

`TERMS_OF_USE_LINK=` for terms of use and

`PRIVACY_LINK=` for privacy information

The default URLs listed above will be redirecting users to generic pages on the labfolder website which provide general information on the use of labfolder.

LDAP Authentication

 Please note that once the LDAP authentication feature is turned on, only users which are registered on the LDAP server will be able to create accounts for labfolder or log on to the system. LDAP authentication is currently possible against the `cn`, but not the `uid`.

Activate LDAP

In order for labfolder to work with an LDAP server a couple of environment variables have to be specified in the `# LDAP Authentication` section in the `server.cnf` file.

1. To turn on LDAP authentication include the flag `FEATURE_LDAP_AUTHENTICATION=true`
2. Specify the URL used to connect to your LDAP server. This should include the protocol and the port. Note two examples are listed below:

```
LDAP_URL=ldap://255.255.255.255:389
```

OR

```
LDAP_URL=ldap://server.company.com:389
```

3. Specify the root distinguished name (DN) of your directory, e.g. `dc=<company>,dc=com`. `LDAP_BASE=<dc=company,dc=com>`
4. Specify the username and password for the search user. The parameters `LDAP_SEARCH_USER_DN` and `LDAP_SEARCH_USER_PASSWORD` are required:

```
LDAP_SEARCH_USER_DN=<CN=search-user,CN=Users,DC=ad,DC=lab  
folder,DC=com>
```

```
LDAP_SEARCH_USER_PASSWORD=***** (The password for your LDAP  
search user)
```

When completing the parameters in your LDAP configuration, do not use braces <> which are used to mark exemplary configurations in the examples above and in the following.

Authenticate with DN

- Specify the allowable pattern(s) of a DN relative to the root DN. Multiple values must be separated by a semicolon. Do not include the root DN. The username will be inserted at the position where {0} occurs in the pattern. Two possible examples are included in what follows:

```
LDAP_USER_DN_PATTERNS=<cn={0},ou=sales,ou=marketing;  
cn={0},ou=development>
```

OR

```
LDAP_USER_DN_PATTERNS=<uid={0},ou=sales,ou=marketing;  
uid={0},ou=development>
```

Authenticate with Attribute

- Set the flag `LDAP_IS_ATTRIBUTE_SEARCH_ENABLED=true`
- Specify the attribute name that the user is supposed to use:

```
LDAP_ATTRIBUTE_SEARCH_NAME=*****
```

Update existing users after LDAP activation

If you would like to activate LDAP and you already have users which have been managed by the labfolder user management, you need to update each user in the user table in the MySQL database as follows:

```
UPDATE user SET ldapUsername = "<ldap-login-id-goes-here>"  
WHERE email LIKE "%<user-email-goes-here>%";
```

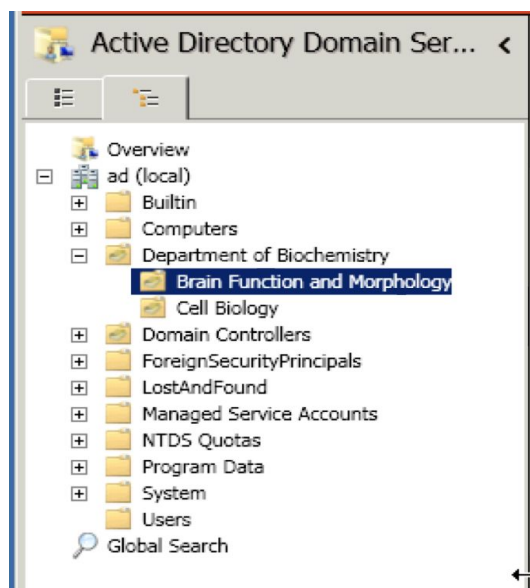
Once LDAP is activated, the login screen will ask for a "Login" which is equivalent to the user name in the LDAP directory (ldapUsername) and will validate against the LDAP server that the user exists and that the password entered during login matches the password stored on the LDAP server.

If LDAP is not activated, the login screen will ask for the e-mail (user-email) and the password stored in the user management system of labfolder.

Examples

Active Directory

The following sample configuration assumes an Active Directory structure which contains users that reside in multiple organization units within the hierarchy.



Configuration

```
# specify server URL
LDAP_URL=ldap://ad.labfolder.com:389

# specify credentials for search user
LDAP_SEARCH_USER_DN=<CN=search-user,CN=Users,DC=ad,DC=labfolder,DC=com>

LDAP_SEARCH_PASSWORD=searchUser123

# specify base DN, to be applied to all patterns
LDAP_BASE=<DC=ad,DC=labfolder,DC=com>

# specify possible DN patterns, DO NOT INCLUDE BASE ABOVE
# {0} will be replaced by the user attempting to bind
LDAP_USER_DN_PATTERNS="CN={0},CN=Users;CN={0},OU=Department of
Biochemistry;CN={0},OU=Brain Function and
Morphology,OU=Department of Biochemistry;CN={0},OU=Cell
Biology,OU=Department of Biochemistry"
```

OpenLDAP

To illustrate another configuration, the following sample configuration will enable the server to connect to a public OpenLDAP server hosted by forumsys:

<http://www.forumsys.com/en/tutorials/integration-how-to/ldap/online-ldap-test-server/>

Note: this is not a production server, and maybe offline from time-to-time. You can use any of the users identified (gauss, euler, euclid, etc) with the password: password.

Configuration

```
# specify server URL
LDAP_URL=ldap://ldap.forumsys.com:389
# specify credentials for search user
LDAP_SEARCH_USER_DN=<cn=read-only-admin,dc=example,dc=com>
LDAP_SEARCH_PASSWORD=password
# specify base DN, to be applied to all patterns
LDAP_BASE=<dc=example,dc=com>
# specify possible DN pattern, DO NOT INCLUDE BASE ABOVE
# {0} will be replaced by the user attempting to bind
LDAP_USER_DN_PATTERNS=<uid={0}>
```

Configuring usage-reporting

Anonymised usage statistics can be sent to labfolder via mail every month. These usage statistics contain the number of registered users on your server, the number of active users on your servers (who have triggered an action in labfolder over the past month) and an activity count per anonymised user ID.

1. Set `CUSTOMER_IDENTIFIER` to the name of your institution.
2. Set `ACTIVE_USER_REPORT_USE_DEFAULT_MAIL_CLIENT=true` if you would like to use the email account that has been configured in the `# Mail client properties` section of the `server.cnf` file. If set to false, labfolder will attempt to connect an external SMTP server via port 25.



If you do not want labfolder to send anonymised user statistics, simply set the variable `FEATURE_ACTIVE_USER_REPORT=false`.

When sending user reports with your own email address, you need to make sure the mail account used is able to send emails outside your domain and network. If `ACTIVE_USER_REPORT_USE_DEFAULT_MAIL_CLIENT=false`, make sure that port 25 can be used by labfolder to contact an external SMTP server.

5. Create the database



On Linux, open a terminal and navigate to the labfolder home directory.

Execute `$./init.sh --user <username> --password <password>`

where `<username>` and `<password>` are your mysql credentials. In case mysql runs on a different server add `"--host <host>"`, where `<host>` is the IP or DNS of your MySQL server.

6. Run labfolder

1. Open a terminal and `cd` into your labfolder installation directory.
2. Run

```
 $ ./labfolder.sh start
```

3. You have to start labfolder if you reboot your server. For automatically launching labfolder on startup, please follow the instructions described in the chapter [Autostart labfolder](#).

Note: labfolder runs on port 9091 over http. We recommend to use *nginx* in case you want to use a SSL certificate.

When starting labfolder, please allow for a few minutes until the labfolder application can be reached via a web browser.

7. Update labfolder

1. Go to the "update" folder in your labfolder installation directory and delete all files in there.
2. Download the update from download link you received together with your license and extract the archive.
3. Replace the „update“ folder in your current installation with the one in the downloaded and extracted archive.



4. Go to your labfolder installation directory and run

```
 $ ./labfolder.sh stop
```

5. Replace the

 labfolder.sh

file in your current installation with the one in the downloaded and extracted archive.

6. Replace the server.cnf file in your current installation with the one in the downloaded and extracted archive. Variables from the old version of the server.cnf file like MySQL credentials can be copied from the old file.
7. If you would like to update the system mails, copy the respective *.html, *.txt and *.subject files from the /files/mail directory from the extracted update package into the /files/mail directory of labfolder.
8. For verification of your labfolder package, copy the file hash.sha256 from the /files directory of the extracted update package into the /files directory of labfolder.
9. Start labfolder as described in the section “Run labfolder”.
10. You can verify your labfolder installation by running the command

 \$./labfolder.sh verify

The script will return OK if the hash sum of your download has been verified.

Activating Apps

XHTML

XHTML export provides a quick and easy way for batch export of data in labfolder. For the default activation of the XHTML App in labfolder, the values in the # Xhtml export section of the *server.cnf* file are set as described below:

```
# Xhtml export
FEATURE_XHTML_EXPORT=true
EXPORT_DOCUMENT_REPOSITORY_TYPE=fileSystem
```

Setting `EXPORT_DOCUMENT_REPOSITORY_TYPE=fileSystem` will result in all files for export being stored in the labfolder `/files` directory, from which they are automatically deleted after 48 hours.

If you would like to store your XHTML export archives on an Amazon S3 web server, set `EXPORT_DOCUMENT_REPOSITORY_TYPE=amazonS3` and add the following parameters to the *server.cnf* file:

```
AMAZON_S3_EXPORT_DOCUMENT_BUCKET_NAME=<name of AS3 bucket>
AMAZON_S3_EXPORT_DOCUMENT_REGION_NAME=<bucket region on AS3>
AMAZON_ACCESS_KEY=<accesskey>
AMAZON_SECRET_ACCESS=<secret>
```

If you want to deactivate XHTML export, set the value `FEATURE_XHTML_EXPORT=false`.

Mendeley

Mendeley is a cloud service for scientific literature management (www.mendeley.com). The labfolder App for Mendeley allows the insertion of citations, the import of PDFs and an upload of exported PDFs. Further information about the Mendeley App can be found in the user manual, which can be downloaded from our website: <https://www.labfolder.com/manuals-tutorials/>.

To activate the Mendeley App, in the #Mendeley section of the server.cnf file, set the feature flag

```
FEATURE_MENDELEY=true
```

Once the App is activated, it becomes visible in the Apps section of labfolder. For the labfolder feature to be fully functional, the labfolder server requires a connection to the internet.



Please note that the Mendeley App is temporarily unavailable in the standalone version 1.20.10 and thus, the feature flag should be set to `FEATURE_MENDELEY=false`.

To activate the Mendeley App, you need to register a ``MENDELEY_CLIENT_ID`` and ``MENDELEY_CLIENT_SECRET`` by signing in to the Developer Program of Mendeley via the website <http://dev.mendeley.com/>. You need to enter a callback url in the following format

```
"your URL"/eln/mendeley/oauthCallback
```

The credentials obtain after the registration process need to be added to the #Mendeley section of the server.cnf file as follows:

```
MENDELEY_CLIENT_ID="Your ID"
```

```
MENDELEY_CLIENT_SECRET="Your secret"
```

```
MENDELEY_CLIENT_CALLBACKURL=http://your  
URL"/eln/mendeley/oauthCallback
```

Mendeley will be activated as soon as labfolder is restarted.

Dropbox

Dropbox is a cloud service which allows the storage and syncing of files (www.dropbox.com). The labfolder App for Dropbox allows the import of files from Dropbox to labfolder and the storage of PDF exports from labfolder in Dropbox. Further information about the Dropbox App can be found in the user manual, which can be downloaded from our website: <https://www.labfolder.com/manuals-tutorials/>.

To activate the Dropbox App, in the #Dropbox section of the server.cnf file, set the feature flag

```
FEATURE_DROPTBOX=true
```

Once the App is activated, it becomes visible in the Apps section of labfolder.

- ⚠ For the use of Dropbox by the users, the Labfolder Standalone Server has to be accessible from the internet.

To set up Dropbox you need a Dropbox Developer Access Key to enable API access. You can create it with any Dropbox account. After creating a Dropbox account create and configure the API access by following these steps:

I. Go to <https://www.dropbox.com/developers/apps/create>

1. Choose an API
 - Dropbox API
2. Choose the type of access you need
 - Full Dropbox
3. Name your app

II. After creation of your app in the Dropbox Developers Section, you can access your Dropbox App settings on <https://www.dropbox.com/developers/apps>

Choosing your app there redirects you to the settings page.

Settings to run with labfolder:

- In the server.cnf file, copy the "App key" as value for the variable `DROPTBOX_CONSUMER_KEY=` and the "App secret" for `DROPTBOX_CONSUMER_SECRET=` into the server.cnf file

- In the Dropbox Developers Section for your selected App, for the variable "OAuth 2 Redirect URIs", set up the custom URL of your server as follows:

```
http(s)://<your_server_url>/eln/dropbox/oauthCallback
```

figshare

- ⚠ The figshare App is temporarily unavailable in standalone version 1.20.10 and thus, the feature flag should be set to `FEATURE_FIGSHARE=false`.

Autostart labfolder

The labfolder application does not automatically start on boot, thus, it is necessary to configure autostart accordingly.



There are multiple ways to configure labfolder autostart after a server reboot. The following sample is designed for Ubuntu.

1. Create a file "labfolderstart" in /etc/init.d/ with the content

```
#!/bin/bash
set -e
```

```
### BEGIN INIT INFO
# Provides:          labfolderstart
# Required-Start:    $docker
# Required-Stop:     $docker
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: labfolder.
# Description:       labfolder.
### END INIT INFO
```

<path to labfolder>/labfolder/labfolder.sh start

2. Create a symlink with the command

```
ln -s /etc/init.d/labfolderstart /etc/rc3.d/S99labfolderstart
```