

ownCloud User Manual

The ownCloud Team

Version 10.5, September 26, 2020

Table of Contents

Introduction.	1
What's New in ownCloud	1
WebUI Overview.	1
Apps.	59
Contacts & Calendar	79
Troubleshooting.	84
Have You Found a Mistake In The Documentation?	87

Introduction

Welcome to ownCloud: your self-hosted file sync and share solution.

ownCloud is an open source file sync and share software for everyone from individuals operating the free ownCloud Server edition, to large enterprises and service providers operating the ownCloud Enterprise Subscription. ownCloud provides a safe, secure, and compliant file synchronization and sharing solution on servers that you control.

You can share one or more files and folders on your computer, and synchronize them with your ownCloud server. Place files in your local shared directories, and those files are immediately synchronized to the server and to other devices using the ownCloud Desktop Sync Client, Android app, or iOS app. To learn more about the ownCloud desktop and mobile clients, please refer to their respective manuals:

- [ownCloud Desktop Client](#)
- [ownCloud Android App](#)
- [ownCloud iOS App](#)

What's New in ownCloud

- Option to hide or expose hidden files in the Web GUI
- Requires to use at least desktop client version 2.6 by default.

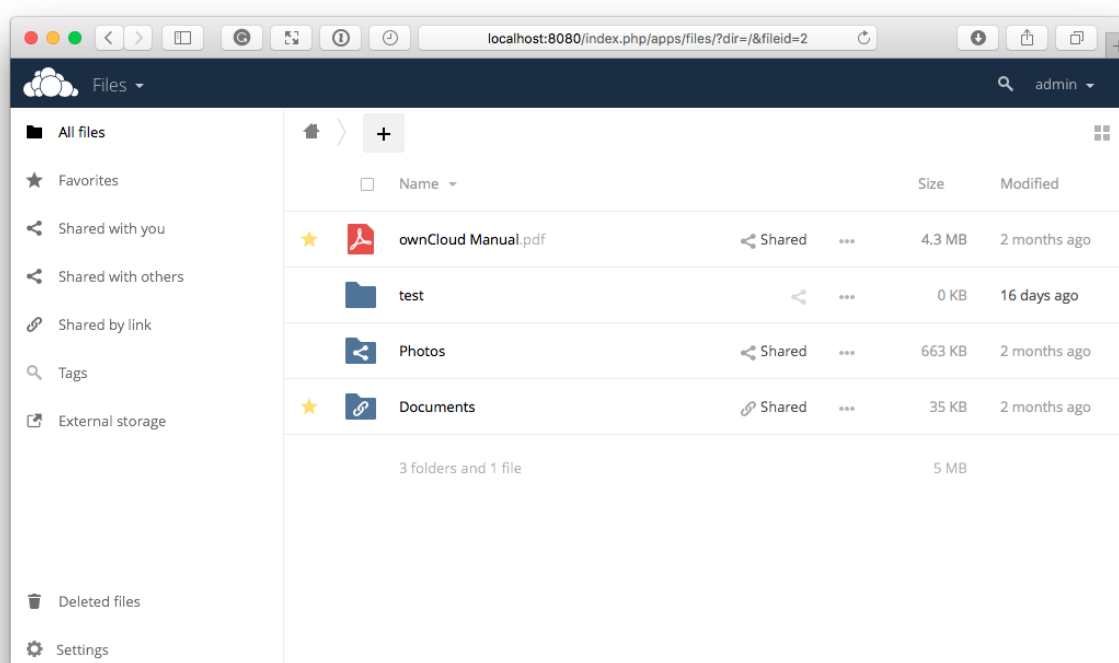
WebUI Overview

Introduction

You can access your files with the ownCloud Web interface, as well as: create, preview, edit, delete, share, and re-share files.



Your ownCloud administrator has the option to disable these features. If any of them are missing on your system ask your server administrator.



File Controls

When you mouseover, or hover over, a file in the Files view, as in the image below, ownCloud displays three file controls. These are:

1. Marking Favorites
2. Sharing Files
3. The Overflow Menu



Marking Favorites

Click the star to the left of the file icon to mark it as a favorite. You can quickly find all of your favorites with the Favorites filter on the left sidebar.



Sharing Files

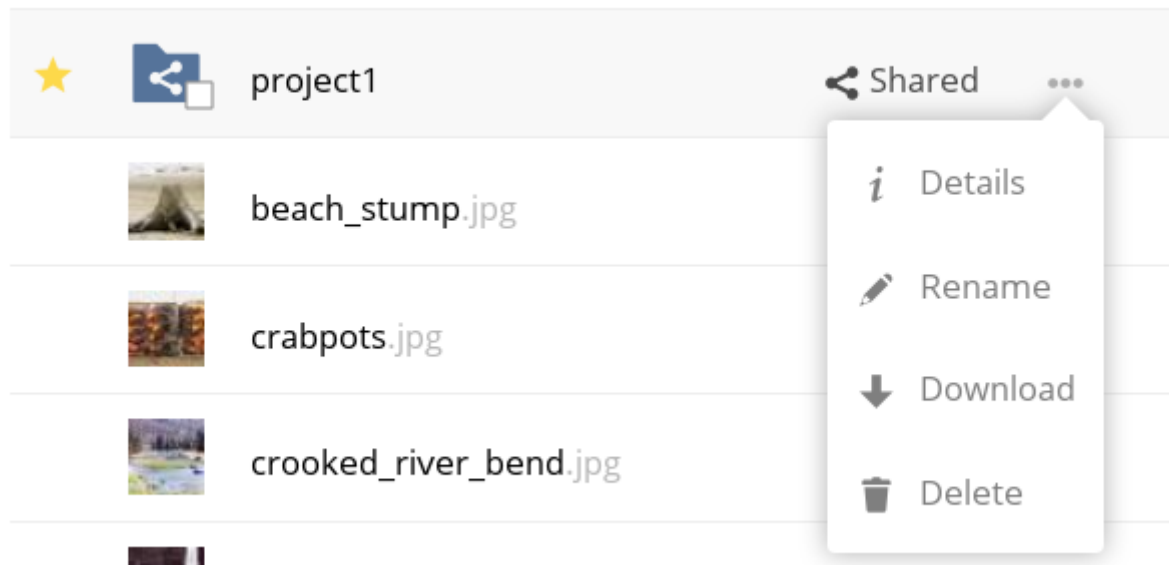
The sharing files control is a shortcut to the file and folder sharing functionality within ownCloud. ownCloud sharing supports:

- Sharing files and folders with users or groups
- Creating public link shares with hyperlinks
- Listing recipients of existing shares
- Deleting existing shares

The Overflow Menu

The *Overflow Menu* allows you to:

- Display File Details
- Rename Files
- Download Files
- Delete Files



Display File Details

When you display details about a file, by clicking [**Details**] in the Overflow Menu, a set of tabs (or views) are available. These are:

View	Description
Details	This shows details about a file, such as its name, size, and when it was created or last updated. To know more, refer to the Details section.
Activity	This shows a history of activity on the file, such as when it was created, updated, and shared. To know more, refer to the Activity section.
Sharing	It's here that shares are managed. To know more, refer to the Sharing Files section.
Version	This shows a history of all the versions of the file. This is not available for folders. To know more, refer to the Version Control section.

You can see an example of the Activity view in the image below.



Rename Files

To rename a file, click **Rename** in the *Overflow Menu*. The file's current name will be displayed in an editable text box. Change the name, click *Enter*, and the file will be renamed. If you don't want to rename it, click *esc* and the file name will be left as is.

Download Files

To download a file, click **Download** in the *Overflow Menu*.

Delete Files

To delete a file, click **Delete** in the *Overflow Menu*.

The Web Interface

You can connect to your ownCloud server using any Web browser; just point it to your ownCloud server and enter your username and password. Supported Web browsers are:

- Edge (current version on Windows 10)
- IE11 or newer (except Compatibility Mode)
- Firefox 60 ESR or newer
- Chrome 66 or newer
- Safari 10 or newer



Some apps like **files_external** or **encryption** will disable the **Stay logged in** checkbox.

Navigating the Main User Interface

By default, the ownCloud Web interface opens to your Files page. You can add, remove, and share files, and make changes based on the access privileges set by you (if you are administering the server) or by your server administrator.

The ownCloud user interface contains the following fields and functions:

- **Apps Selection Menu:** Located in the upper left corner, click the arrow to open a dropdown menu to navigate to your various available apps.
- **Apps Information Field:** Located in the left sidebar, this provides filters and tasks associated with your selected app. For example, when you are using the Files apps you have a special set of filters for quickly finding your files, such as files that have been shared with you, and files that you have shared with others. You'll see different items for other apps.
- **Application View:** The main central field in the ownCloud user interface. This field displays the contents or user features of your selected app.
- **Navigation Bar:** Located over the main viewing window (the Application View), this bar provides a type of breadcrumbs navigation that enables you to migrate to higher levels of the folder hierarchy up to the root level (home).
- **New Button:** Located in the Navigation Bar, the **[New]** button enables you to create new files, new folders, or upload files.



You can also drag and drop files from your file manager into the ownCloud Files Application View to upload them to ownCloud. Currently, the only Web browsers that support drag-and-drop folders are Chrome and Chromium.

- **Search Field:** Click on the **[magnifier]** in the upper right hand corner of to search for files.
- **Personal Settings Menu:** Click on your ownCloud **[username]**, located to the right of the Search field, to open your Personal Settings dropdown menu. Your Personal page provides the following settings and features:

-
- Links to download desktop and mobile apps
 - Re-run the First Run Wizard
 - Server usage and space availability
 - Password management
 - Name, email, and profile picture settings
 - Manage connected browsers and devices
 - Group memberships
 - Interface language settings
 - Manage notifications
 - Federated Cloud ID
 - Social media sharing buttons
 - SSL certificate manager
 - ownCloud Version information

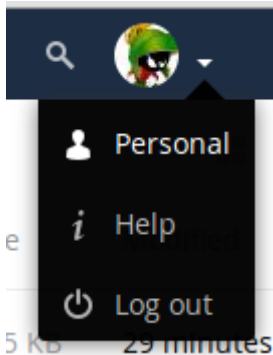
See userpreferences section to learn more about these settings.

Personal Settings

As a user, you can manage your personal settings. To access them:

1. Click on your username in the top, right-hand corner of the WebUI of your ownCloud instance.

The Personal Settings Menu opens.



Personal Settings Menu

2. Then, click **Personal** from the drop down menu.

You are using 698 KB of Unlimited

Profile picture



png or jpg, max. 20 MB

Full name

bernie madhatter

Email

bernie@example.com

For password recovery and notifications

Groups

You are member of the following groups:
graphics, users, video

Password

Current password

New password 

Change password



The options listed in the Personal Settings page depend on the applications that are enabled by the administrator.

Some of the features you will see include the following:

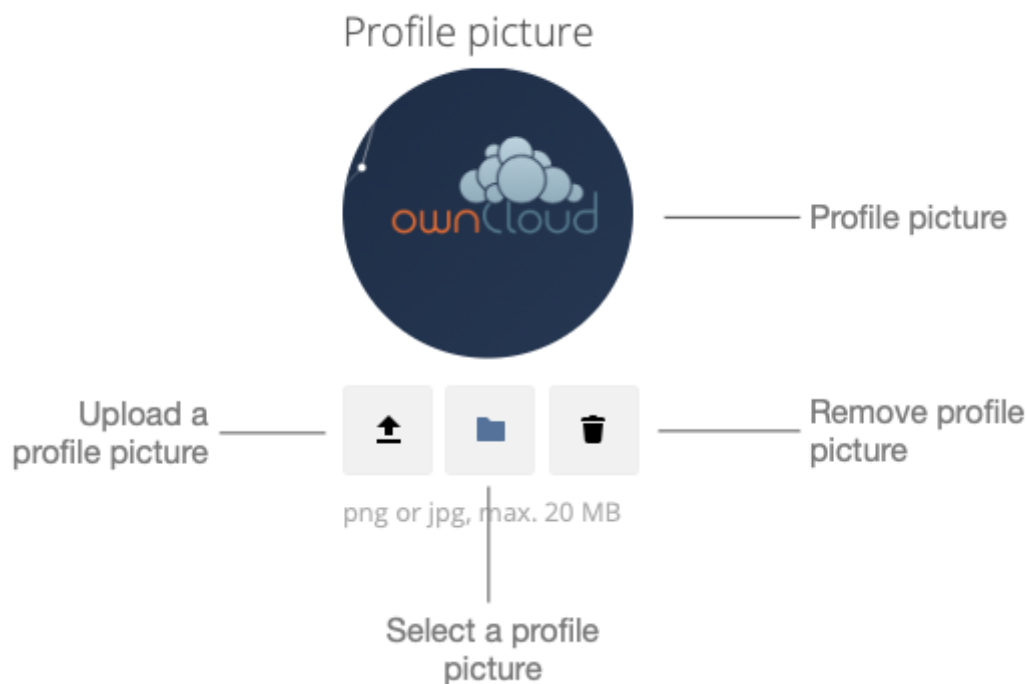
- Usage and available quota
- Manage your profile picture
- Full name You can make this anything you want, as it is separate from your ownCloud login name, which is unique and cannot be changed
- Email address
- Lists your Group memberships
- Manage your password
- Choose the language for your ownCloud interface
- Links to desktop and mobile apps
- Manage your Activity stream and notifications
- Default folder to save new documents to
- Your Federated sharing ID
- Social sharing links
- ownCloud version

General Settings

In the general settings, you can do the following:

- Set:
 - Your profile picture
 - [The language for your user account](#)
- Update your personal data:
 - Full name
 - Email address
 - [Password](#)
- View:
 - The amount of storage you have used
 - The groups that you are a member of
 - Your federated cloud id
 - The current version of ownCloud
- Download the desktop, android, and iOS app

Managing Your Profile Picture



Changing Your Profile Picture

There are two options to set your profile picture:

- You can upload an image; or
- Select an existing profile picture

Upload a Profile Picture


To upload an image click the "Upload new"  button, which opens a file browser, through which you can choose an image to upload. After you've chosen an image, you will then be able to crop the uploaded image, if required, to just the segment of the

image that you want to use for your profile picture. You can see an example in the screenshot below.



Figure 1. Crop the new profile picture

When you're happy with the image, click **[Choose as profile picture]**, and the image will be set as your profile picture.

 the image can be a maximum of 20MB in size.

Select a Profile Picture


To select a profile picture, click the *Select from Files*  button and the profile picture chooser will open. From there, pick the profile picture that you want and click **[Choose]**.



Figure 2. Choose a new profile picture

Remove Your Profile Picture

To remove a custom profile picture, click the *Remove Profile*  button. After that, the image will revert to the default image, present when you first logged in.

Update Your Full Name

To update (or change) your full name, amend the existing text in the text box below “Full name”. After a few seconds, your full name will be auto-saved.



Figure 3. Successfully updated full name

If the full name cannot be changed, you will see a notification, as in the example below.

Full name

Unable to change full name

Figure 4. Problem updating full name

Update Your Email Address

To update your email address, change the address in the text box below “Email” and click [Set email].


Update Your Password

Password

Change password

Figure 5. Update current password.

To change your password, under Password, enter your current password in the first password field (with the placeholder text “Current password”) and your new password in the second password field (with the placeholder text “New password”), and then click [Change password].



Password changes automatically log out **all** connected browsers/devices.

Set the Language

Language

English

Help translate

Figure 6. Change profile’s language

To change the language for your user account, pick the desired language from the dropdown under Language. The new language will be auto-saved shortly after you select it.

External Storage

If your ownCloud administrator has enabled External Storage, and how they have configured external storage, then in this section you will be able to add one or multiple external storages.

Personal

General

Storage

Sharing

External Storage

Folder name	External storage	Authentication	Configuration
<input type="text" value="Folder name"/>	Add storage		

Figure 7. User Storage Settings

To configure one or more external storages, please refer to [the External Storage](#)

documentation.

Sharing

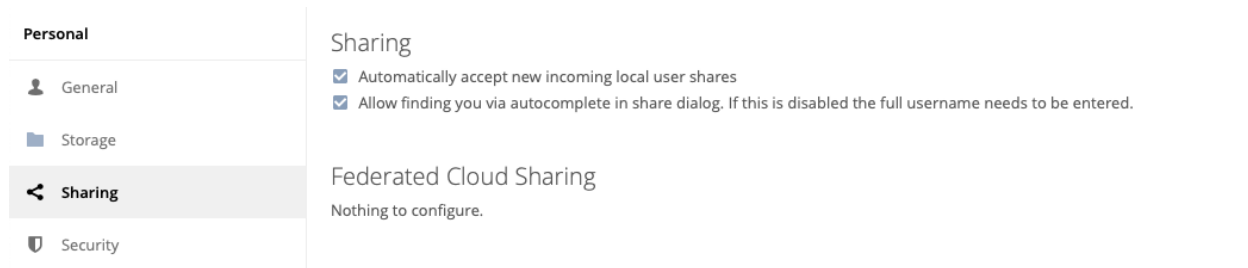


Figure 8. User Sharing Settings

This section allows you to set whether or not you want to:

- Automatically accept new incoming local user shares
- Allow finding you via autocomplete in share dialog.

To enable either, check the respective checkbox. This section also allows for configuring [Federated Cloud Sharing](#).

Security

The security settings page allows for:

- Managing CORS white-listed domains
- Viewing and deleting user sessions
- Creating new app passcodes

CORS White-listed Domains

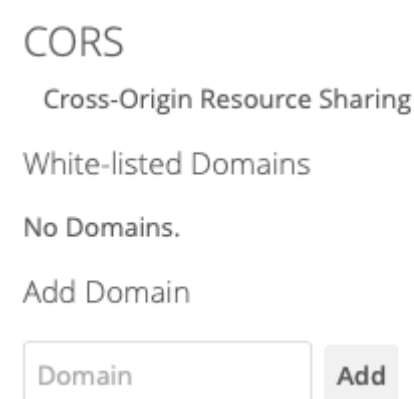


Figure 9. CORS (Cross-origin Resource Sharing) configuration section


The [CORS \(Cross-origin resource sharing\)](#) white-listed domains section lists zero or more domains which the ownCloud instance is allowed to request resources from, in addition to the current domain, for your user account.

By default, as in the screenshot above, no domains will be listed. If you want to add one, or more, add them, one at a time, in the Domain text field, and click **[Add]**. You will then seem list listed, as in the screenshot below. To remove one, or more, click the trashbin icon next to the relevant domain name.

CORS

Cross-Origin Resource Sharing

White-listed Domains

Domain
http://localhost 

Add Domain

Figure 10. CORS Configuration Section with white-listed domains

You will then be prompted to confirm if you want to remove the domain. If you do, click **[Yes]**. If you do not, click **[No]**.

Sessions

Sessions

These are the web, desktop and mobile clients currently logged in to your ownCloud.




Browser	Most recent activity	
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/6...	seconds ago	
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/6...	5 hours ago	
Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:69.0) Gecko/201001...	12 days ago	
Mozilla/5.0 (Macintosh; Intel Mac OS X 10.14; rv:69.0) Gecko/20...	14 days ago	

Figure 11. Sessions section

The sessions section, which you can see an example of below, lists all your current user sessions, across web, desktop, and mobile clients. Specifically, it lists the browser user agent string and the time of the most recent activity. If you want to log a session out, then click the Disconnect button at the far right of the relevant session.

App Passwords / Tokens

App passwords / tokens

App passwords or tokens are passcodes that give an app or device permissions to access your ownCloud account. Use them as a security measure to hide your actual password which you may only want to use for web interface login.

Figure 12. App password / tokens section

This section lets you give an app or device permissions to access your ownCloud account. App passwords are a security measure which let you hide your actual password. To create one, insert the app name in the App name text field, and click **[Create new app passcode]**.

1. Create new app password / token image::personal-settings/security/create-new-app-password-token.png[]

As you can see in the screenshot above, a username and password/token will be

generated, and the app will be listed in the apps list in this section.



Make sure you either securely store the username and password / token or ensure that the receiver does, because once you click **[Done]** the username and password / token will longer be discoverable.

If you want to revoke access for a device or app, click the trash bin icon next to its name in the apps list.



No confirmation of revocation is requested. Once you click the trash bin icon, the apps access is revoked.

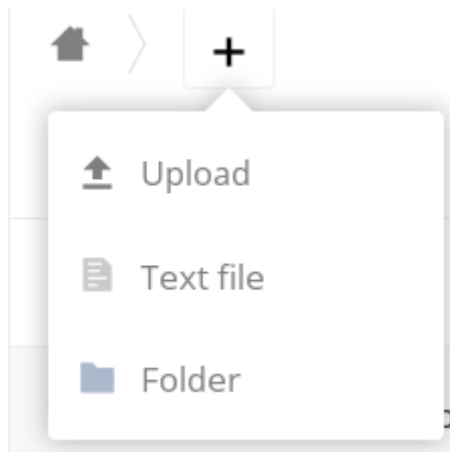
Navigating the WebUI

Introduction

Navigating through folders in ownCloud is as simple as clicking on a folder to open it and using the back button on your browser to move to a previous level. This section walks you through how to navigate the ownCloud UI.

Create and Upload Files and Directories

At the top of the Files view is a navigation bar. This contains links to uploading new files, and creating new files and folders.



To upload or create new files or folders directly in an ownCloud folder click on the **[New]** button in the navigation bar (this is the **+** button). There, as in the image above, you can see links to:

- **[Upload a new file]** This uploads files from your computer into ownCloud. You can also upload files by dragging and dropping them from your file manager.
- **[Create a new text file]** This creates a new text file and adds the file to your current folder.
- **[Create a new folder]** This creates a new folder in the current folder.

Select Files or Folders

<input type="checkbox"/>	Name ▾		Size	Modified
	ownCloud Manual.pdf	Shared ...	4.3 MB	2 months ago
	test	...	0 KB	16 days ago
	Photos	Shared ...	663 KB	2 months ago
	Documents	Shared ...	35 KB	2 months ago

You can select one or more files or folders by hovering over them (as in the image below) and clicking on their checkboxes. To select all files in the current directory, click on the checkbox located at the top of the files listing.

<input checked="" type="checkbox"/>	3 folders and 1 file		Download	5 MB	Delete
	ownCloud Manual.pdf	Shared ...	4.3 MB	2 months ago	
	test	...	0 KB	16 days ago	
	Photos	Shared ...	663 KB	2 months ago	
	Documents	Shared ...	35 KB	2 months ago	

When you select multiple files, you can delete all of them, or download them as a ZIP file by using the **[Delete]** or **[Download]** buttons that appear at the top.

	If the [Download] button is not visible, the administrator has disabled this feature.
--	--

Filter the Files View

The left sidebar on the Files page contains several filters for quickly sorting and managing your files.

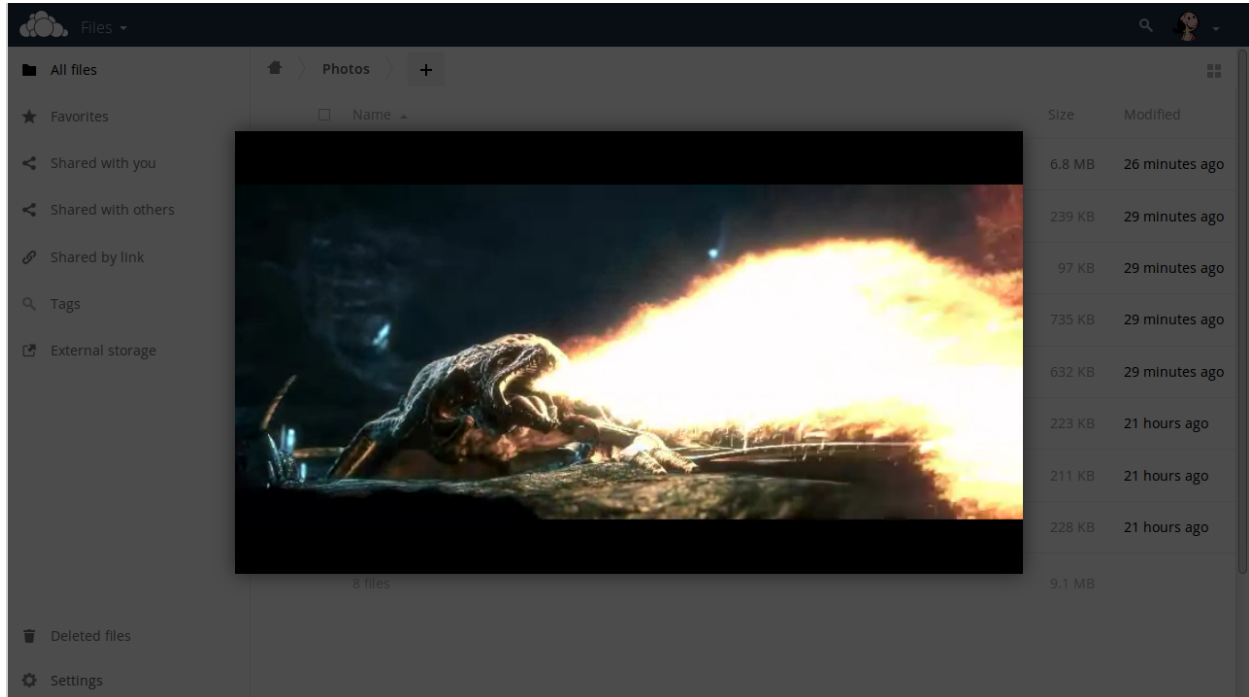
Option	Description
All files	The default view; displays all files that you have access to
Favorites	Files or folders marked with the yellow star
Shared with you	Displays all files shared with you by another user or group
Shared with others	Displays all files that you have shared with other users or groups
Shared by link	Displays all files that are shared by you via public link
External Storage	Files that you have access to on external storage devices and services such as Dropbox, Google, and Amazon S3

Move Files

You can move files and folders by dragging and dropping them into any directory.

Play Videos

You can play videos in ownCloud with the Media Viewer app, by clicking once on the file. Please note, video streaming by the ownCloud Media Viewer depends on your web browser and the video's format.



If your ownCloud administrator has enabled video streaming, and it doesn't work in your Web browser, it may be a browser-related issue. See [Mozilla's Browser Compatibility Guide](#) for supported multimedia formats in Web browsers.

Settings

The **Settings** gear icon, in the lower left-hand corner of the ownCloud window, allows you to show or hide hidden files in your ownCloud Web interface. These are also called dotfiles, because they are prefixed with a dot, e.g. `.mailfile`.

The dot tells your operating system to hide these files in your file browsers, unless you choose to display them. Usually, these are configuration files, so having the option to hide them reduces clutter.



☐ Show hidden files

Preview Files

ownCloud can display thumbnail previews for *images*, *MP3 covers*, and *text files*, if this is enabled by your server administrator. You can also display *uncompressed text*, *OpenDocument*, *videos*, and *image* files in the ownCloud embedded viewers by clicking on the file name. There may be other file types you can preview if your ownCloud administrator has enabled them. If ownCloud cannot display a file, it will start a download process and downloads the file to your computer.

Comments

Introduction

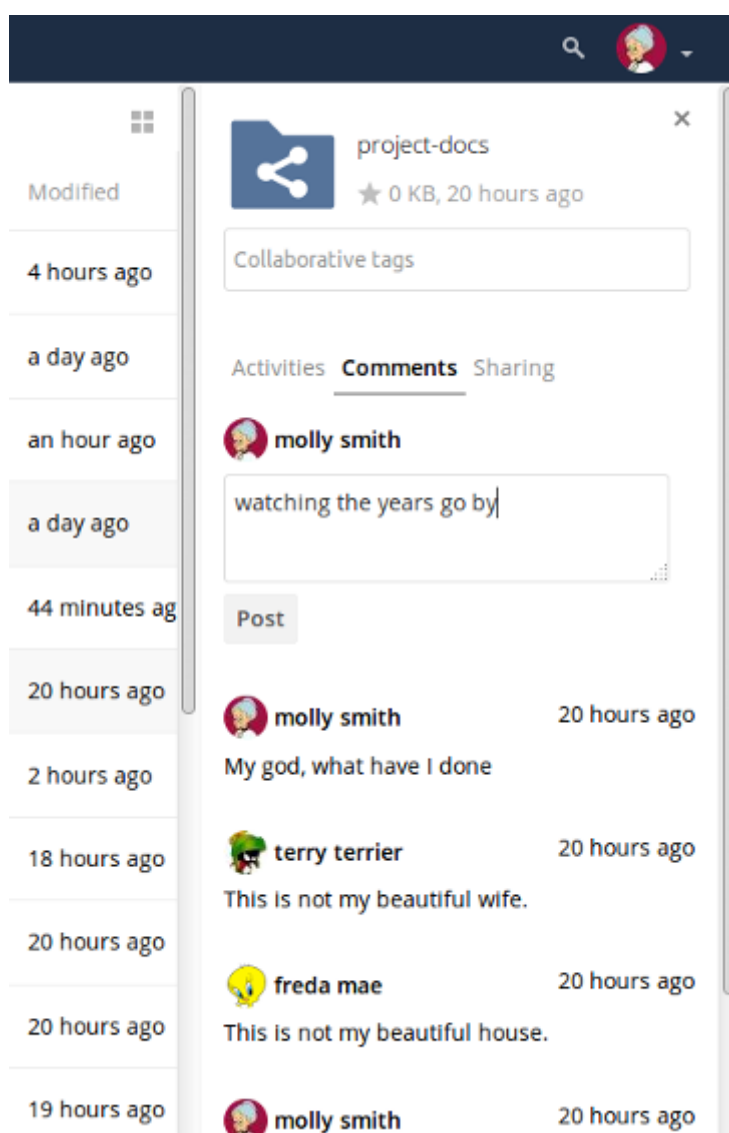
In ownCloud, you can add one or more comments on both files and folders. This section describes how to add, edit, and delete comments.



You can navigate directly to the comments pane for a file by using the URL: [https://your.owncloud.domain/f/?<\\$fileId>?details=commentsTabView](https://your.owncloud.domain/f/?<$fileId>?details=commentsTabView), and substituting **<\$fileId>** for the file's id.

Add Comments

Use the Details view, in [The Overflow Menu](#), to add and read comments on any file or folder. Comments are visible to everyone who has access to the file or folder. To add a comment, as in the example below, click the **[Comments]** tab in the Details view, write a comment in the New Comment field, and click **[Post]**.



Edit Comments

To edit an existing comment on a file or folder, hover the mouse over the comment and you will see a pencil icon appear. By clicking on the pencil, the **[Edit Comment]** field will appear, pre-filled with the comment text. Change the text as necessary and click **[Save]**. If you change your mind, just click **[Cancel]**.

Delete Comments

To delete an existing comment on a file or folder, as with editing comments, hover the mouse over the comment and you will see a pencil icon appear. Click the pencil, and a rubbish bin icon appears on the far right-hand side of the comment author's name, above the [Edit Comment] text field. Click the [rubbish bin], and the comment will be deleted after a few seconds.

Custom Groups

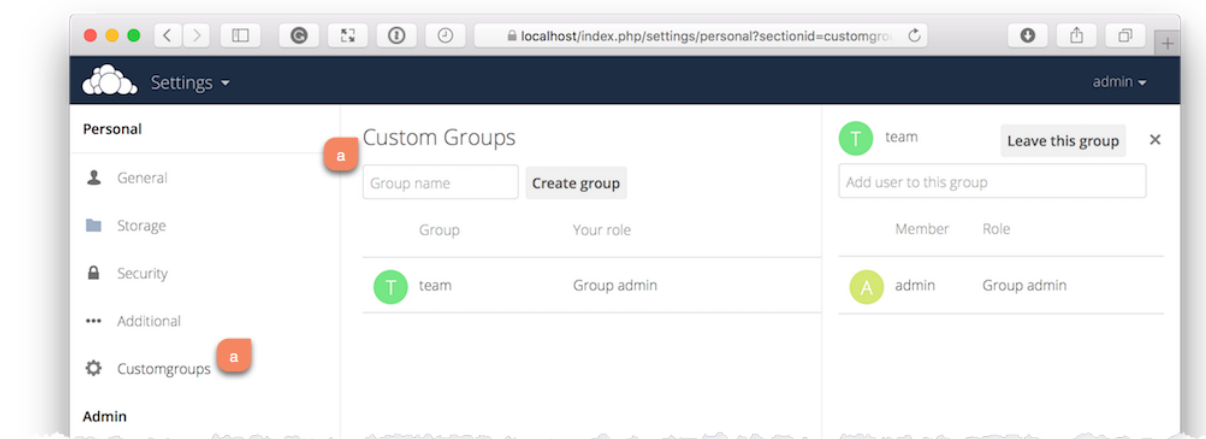
Introduction

In previous versions of ownCloud, if you wanted to share a file or a folder with more than one person, you had to share it either with many people individually, or share to one or more groups. However, you could only share with groups which your ownCloud administrator had already created.


This wasn't the most efficient way to work. To address that, as of ownCloud 10.0, you can now create your own groups on-the-fly, through a feature called "Custom Groups". Here's how to use it.

Creating Custom Groups

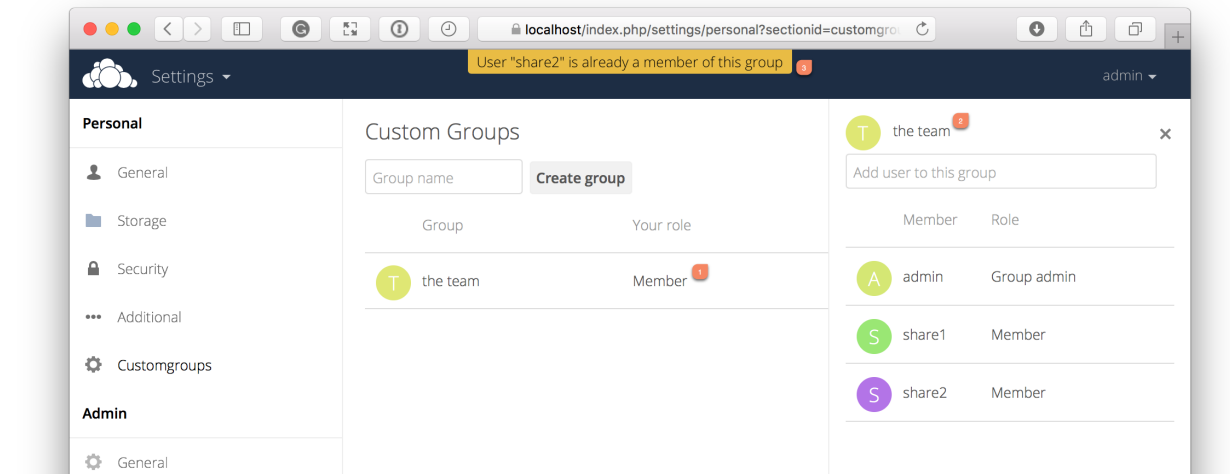
Assuming that your ownCloud administrator's already **enabled custom groups**; under the admin menu, in the top right-hand corner, click [Settings] (1). Then, in the main menu on the settings page, in "**Personal**" section, click the option: [Customgroups] (2). This will take you to the "**Custom Groups**" admin page.



To create a new custom group, in the text field at the top where you see the placeholder text: "**Group name**", add the group name and click [Create group]. After a moment or two, you'll see the new custom group appear in the groups list.

	<p>Please be aware of two things:</p> <ol style="list-style-type: none">1. Custom groups are visible only to members of the group, but not to anyone outside the group; and2. ownCloud administrators can see and modify all custom groups of an instance.
---	---

Managing Group Members



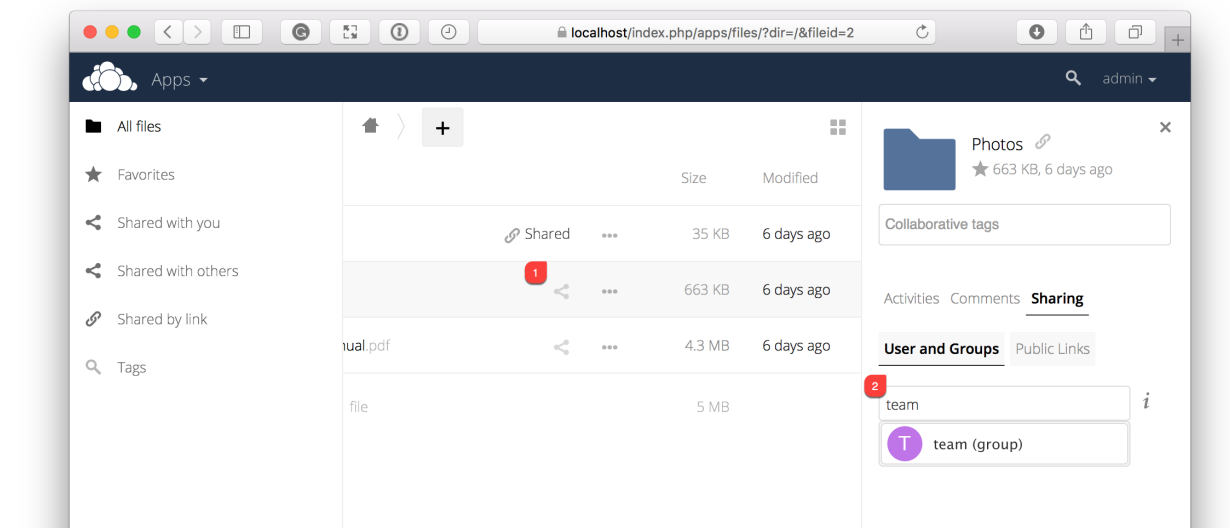
To add or remove users in a custom group, click your role (1), which will likely be "**Member**" (at least at first), and you'll see a panel appear on the right-hand side listing the group's users and their roles. In the "**Add user to this group field**" at the top of the panel (2), start typing the name of the user that you want to add.

After a moment or two, you'll see a list of users that match what you've typed appear (if there are any) in a popup list. Click the one that you want, and they'll be added to the group. Finally, you'll see a confirmation at the top of the page (3), indicating that the user's been added to the custom group.



Members can only use a group for sharing, whereas group admins can manage a group's members, change a group's name, change members' roles, and delete groups.

Sharing with the Group

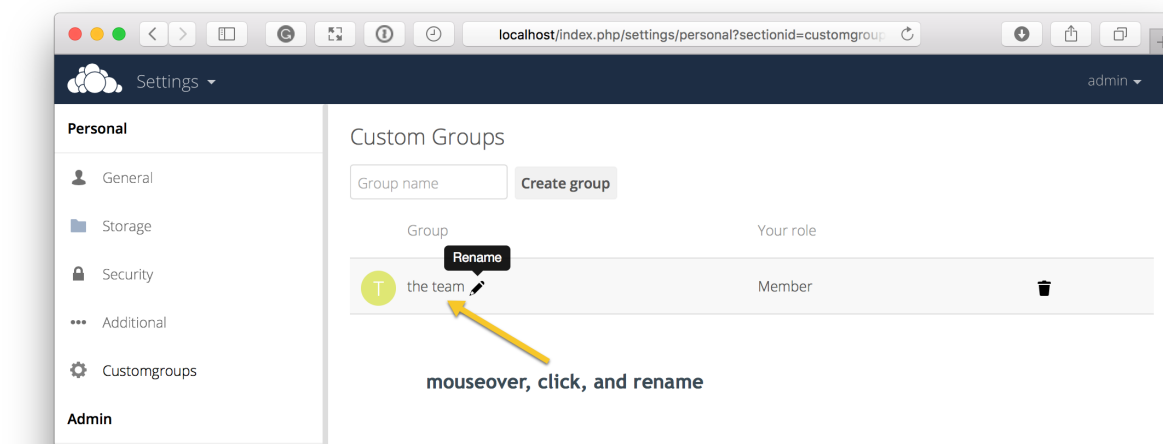


To share a file or folder with your custom group, open the sharing panel (1). Then, in the "**User and Groups**" field (2), type part of the name of the custom group and wait a moment or two.

The name of the group should be displayed in a popup list, which you can see in the screenshot above. Click on it, and the file or folder will then be shared with your custom group with all permissions initially set.

Changing Group Names

If you want to change the name of the custom group, mouseover the group's name in the custom groups list, where you will see a pencil appear to the right of the existing name. Click it, and a text field will appear, pre-populated with the existing name. Change the name and click enter, and the name will be changed.



Files

This section covers how to work with and user files when using ownCloud.

Accessing ownCloud Files Using WebDAV

Introduction

ownCloud fully supports the WebDAV protocol, and you can connect and synchronize with your ownCloud files over WebDAV. In this chapter you will learn how to connect Linux, Mac OS X, Windows, and mobile devices to your ownCloud server via WebDAV. Before we get into configuring WebDAV, let's take a quick look at the recommended way of connecting client devices to your ownCloud servers.

ownCloud Desktop and Mobile Clients

The recommended method for keeping your desktop PC synchronized with your ownCloud server is by using the [ownCloud Desktop Client](#). You can configure the ownCloud client to save files in any local directory you want, and you choose which directories on the ownCloud server to sync with. The client displays the current connection status and logs all activity, so you always know which remote files have been downloaded to your PC, and you can verify that files created and updated on your local PC are properly synchronized with the server.

The recommended method for syncing your ownCloud server with Android and Apple iOS devices is by using the [ownCloud mobile apps](#).

To connect to your ownCloud server with the **ownCloud** mobile apps, use the base URL and folder only:

```
example.com/owncloud
```

In addition to the mobile apps provided by ownCloud, you can use other apps to connect to ownCloud from your mobile device using WebDAV. [WebDAV Navigator](#) is a good (proprietary) app for [Android devices](#), [iPhones](#), and [BlackBerry devices](#). The URL to use on these is:

example.com/owncloud/remote.php/webdav

WebDAV Configuration

If you prefer, you may also connect your desktop PC to your ownCloud server by using the WebDAV protocol rather than using a special client application. Web Distributed Authoring and Versioning (WebDAV) is a Hypertext Transfer Protocol (HTTP) extension that makes it easy to create, read, and edit files on Web servers. With WebDAV you can access your ownCloud shares on Linux, Mac OS X and Windows in the same way as any remote network share, and stay synchronized.



In the following examples, You must adjust **example.com/** to the URL of your ownCloud server installation.

Accessing Files Using Linux

You can access files in Linux operating systems using the following methods.

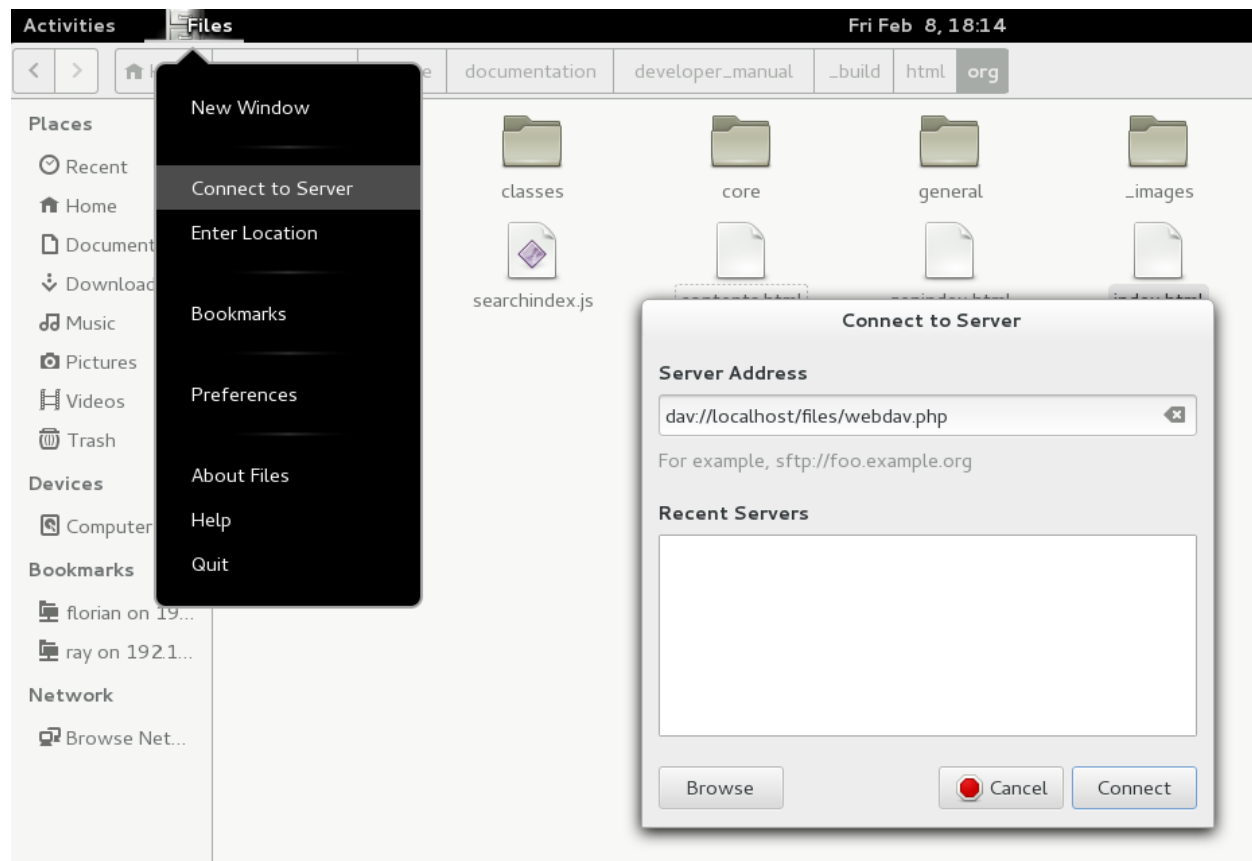
Nautilus File Manager

Use the **davs://** protocol to connect the Nautilus file manager to your ownCloud share:

davs://example.com/owncloud/remote.php/webdav



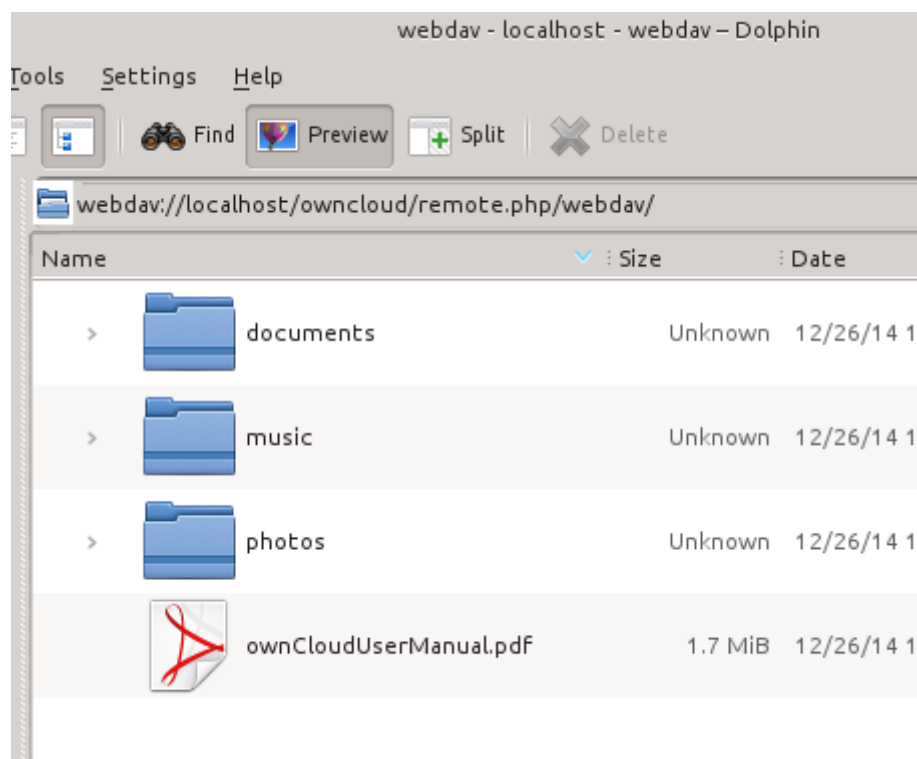
If your server connection is not HTTPS-secured, use **dav://** instead of **davs://**.



Accessing Files with KDE and Dolphin File Manager

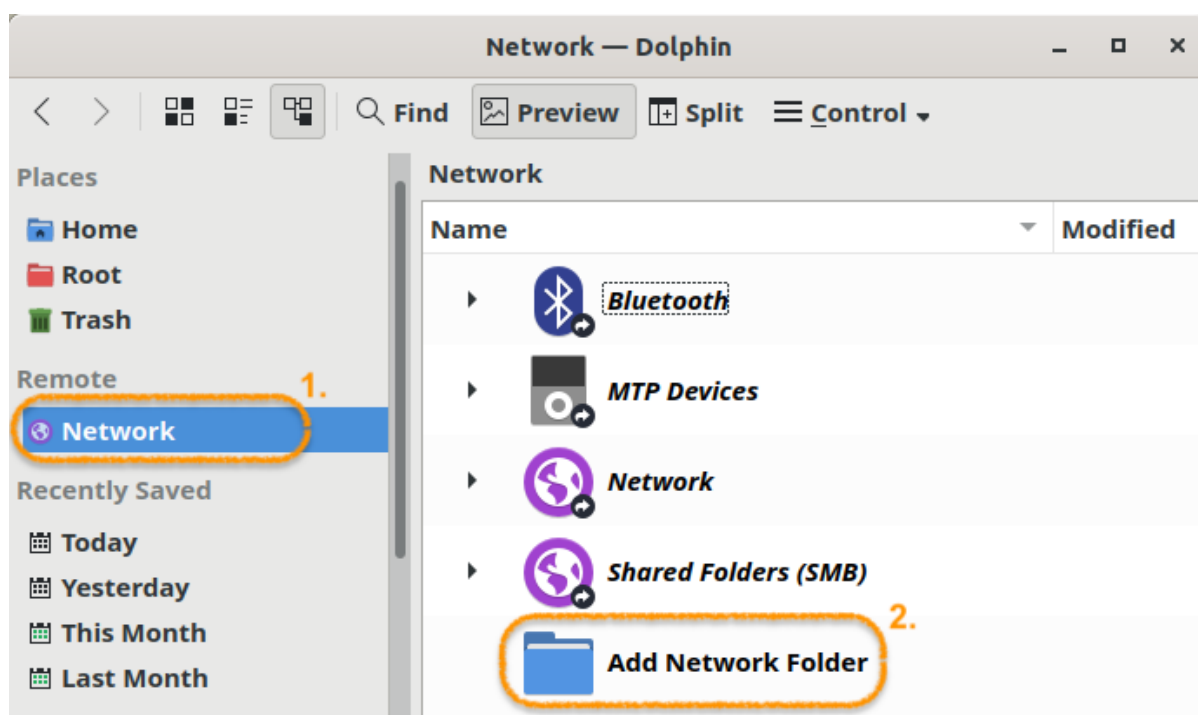
To access your ownCloud files using the Dolphin file manager in KDE, use the **webdav://** protocol:

```
webdav://example.com/owncloud/remote.php/webdav
```



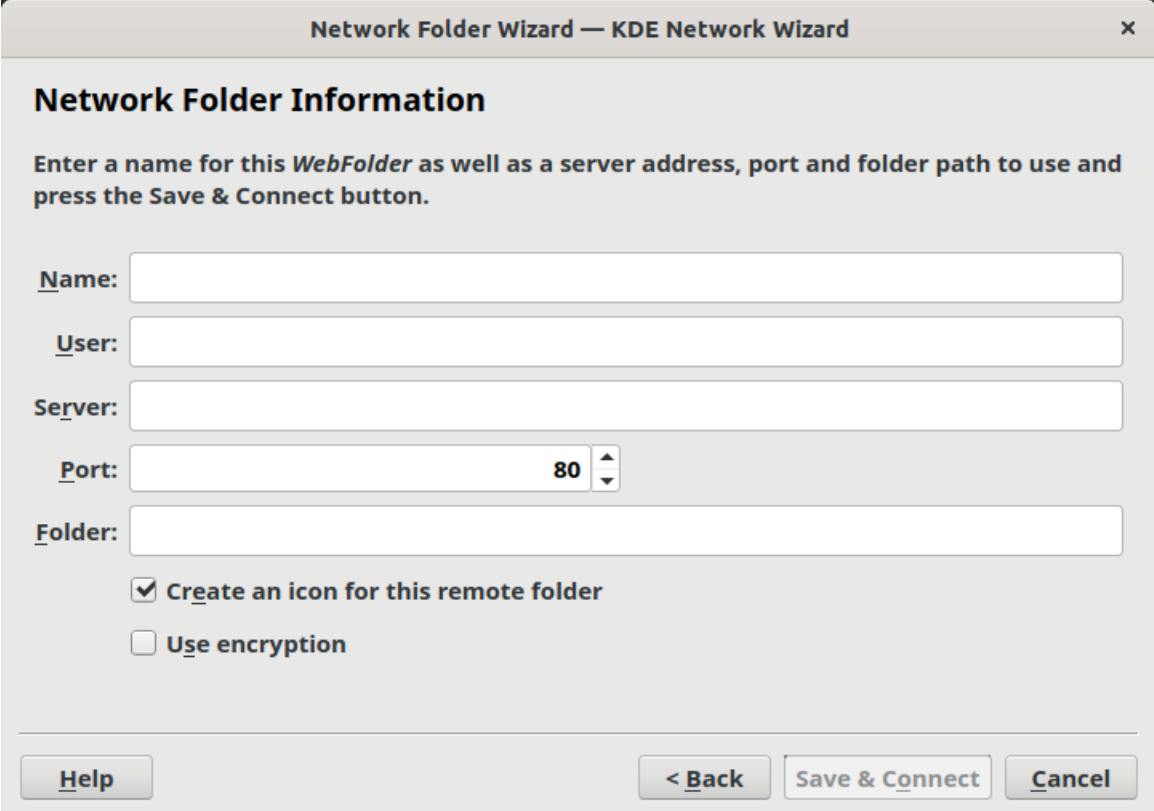
You can create a permanent link to your ownCloud server:

1. Open Dolphin and click **[Network]** in the left hand column.



2. Click on the icon labeled **[Add a Network Folder]**.
The resulting dialog should appear with WebDAV already selected.

3. If WebDAV is not selected, select it.
4. Click **[Next]**.
5. Enter the following settings:
 - **Name:** The name you want to see in the **Places** bookmark, for example ownCloud.
 - **User:** The ownCloud username you used to log in, for example admin.
 - **Server:** The ownCloud domain name, for example **example.com** (without **https://** or **http://**).
 - **Folder:** Enter the path **owncloud/remote.php/webdav**.



6. (Optional) Check the **[create]** icon checkbox for a bookmark to appear in the **Places** column.
7. (Optional) Provide any special settings or an SSL certificate in the **[Port & Encrypted]** checkbox.

Creating WebDAV Mounts on the Linux Command Line

You can create WebDAV mounts from the Linux command line. This is useful if you prefer to access ownCloud the same way as any other remote filesystem mount. The following example shows how to create a personal mount and have it mounted automatically every time you log in to your Linux computer.

1. Install the **davfs2** WebDAV filesystem driver, which allows you to mount WebDAV shares just like any other remote filesystem. Use this command to install it on Debian/Ubuntu:

```
sudo apt-get install davfs2
```

2. Use this command to install it on CentOS, Fedora, and openSUSE:

```
sudo yum install davfs2
```

3. Add yourself to the **davfs2** group (this will be effective after the next login):

```
sudo usermod -aG davfs2 <username>
```

4. Then create an **owncloud** directory in your home directory for the mountpoint, and **.davfs2/** for your personal configuration file:

```
mkdir ~/owncloud  
mkdir ~/.davfs2
```

5. Copy **/etc/davfs2/secrets** to **~/.davfs2**:

```
sudo cat /etc/davfs2/secrets > ~/.davfs2/secrets
```

6. Make the permissions read-write owner only:

```
chmod 600 ~/.davfs2/secrets
```

7. Add your ownCloud login credentials to the end of the **secrets** file, using your ownCloud server URL and your ownCloud username and password:

```
/home/<username>/owncloud <username> <password>
```

8. Add the mount information to **/etc/fstab**:

```
https://example.com/owncloud/remote.php/webdav  
/home/<username>/owncloud davfs user,rw,auto 0 0
```

9. Then test that it mounts and authenticates by running the following command. If you set it up correctly you won't need root permissions:

```
mount ~/owncloud
```

10. You should also be able to unmount it:

```
umount ~/owncloud
```

Now every time you login to your Linux system your ownCloud share should automatically mount via WebDAV in your **~/owncloud** directory. If you prefer to mount it manually, change **auto** to **noauto** in **/etc/fstab**.

Known Issues

Problem: Resource Temporarily Unavailable

Solution

If you experience trouble when you create a file in the directory, edit `/etc/davfs2/davfs2.conf` and add:

```
use_locks 0
```

Problem: Certificate Warnings

Solution

If you use a self-signed certificate, you will get a warning. To change this, you need to configure `davfs2` to recognize your certificate. Copy `mycertificate.pem` to `/etc/davfs2/certs/`. Then edit `/etc/davfs2/davfs2.conf` and uncomment the line `servercert`. Now add the path of your certificate as in this example:

```
servercert /etc/davfs2/certs/mycertificate.pem
```

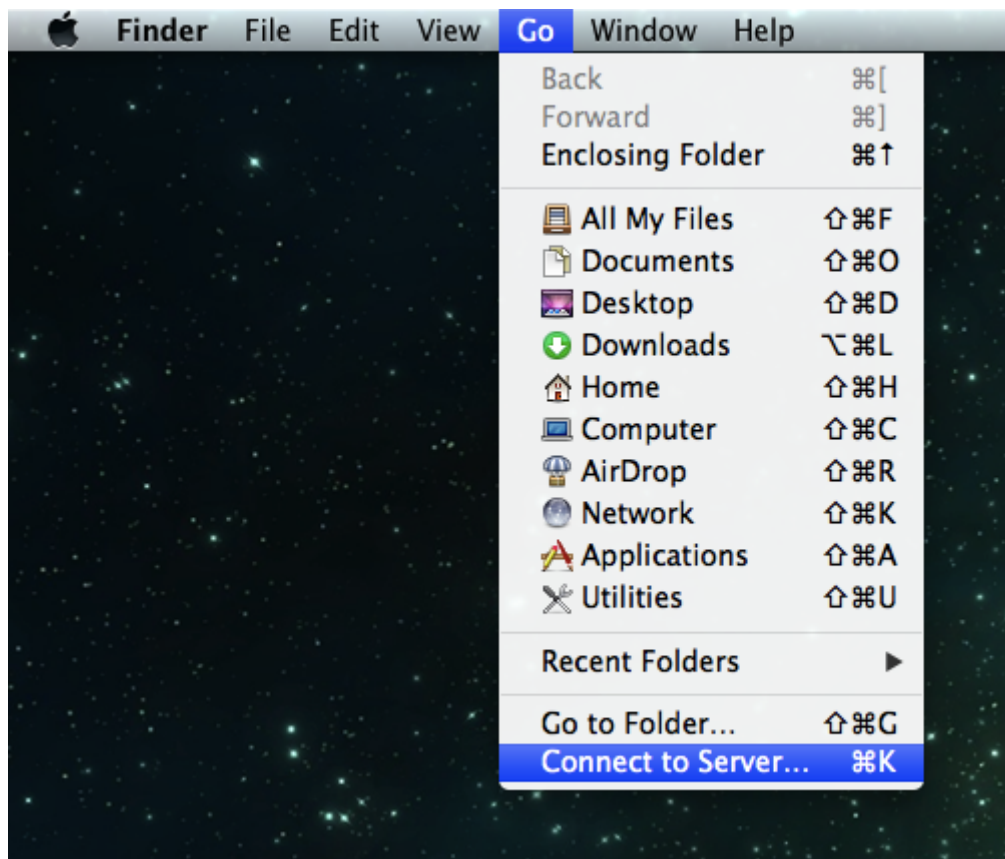
Accessing Files Using Mac OS X



The Mac OS X Finder suffers from a [series of implementation problems](#) and should only be used if the ownCloud server runs on **Apache** and **mod_php**. You can use a tool like [ocsmount](#) to mount without those issues.

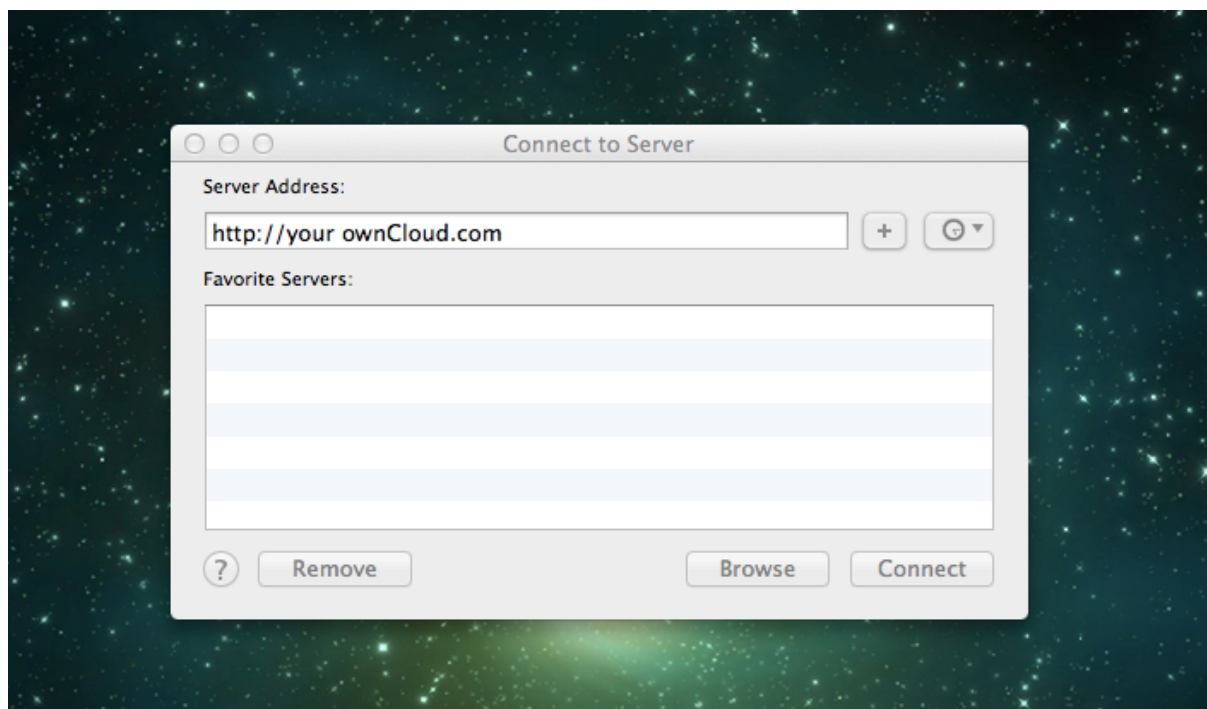
To access files through the Mac OS X Finder:

1. Choose **Go > Connect to Server**.
The "Connect to Server" window opens.
2. Specify the address of the server in the **Server Address** field.



For example, the URL used to connect to the ownCloud server from the Mac OS X Finder is:

```
https://example.com/owncloud/remote.php/webdav
```



3. Click **[Connect]**.
The device connects to the server.

For added details about how to connect to an external server using Mac OS X, check the [wikihow documentation](#)

Accessing Files Using Microsoft Windows

It is best to use a suitable WebDAV client from the [WebDAV Project](#) page .

If you must use the native Windows implementation, you can map ownCloud to a new drive. Mapping to a drive enables you to browse files stored on an ownCloud server the way you would files stored in a mapped network drive.

Using this feature requires network connectivity. If you want to store your files offline, use the ownCloud Desktop Client to sync all files on your ownCloud to one or more directories of your local hard drive.



Prior to mapping your drive, you must permit the use of Basic Authentication in the Windows Registry. The procedure is documented in [KB841215](#) and differs between Windows XP/Server 2003 and Windows Vista/7. Please follow the Knowledge Base article before proceeding, and follow the Vista instructions if you run Windows 7.

Mapping Drives With the Command Line

The following example shows how to map a drive using the command line. To map the drive:

1. Open a command prompt in Windows.
2. Enter the following line in the command prompt to map to the computer Z drive, where `<drive_path>` is the URL to your ownCloud server:

```
net use Z: https://<drive_path>/remote.php/webdav /user:youruser  
yourpassword
```

Example:

```
net use Z: https://example.com/owncloud/remote.php/webdav /user:youruser  
yourpassword
```

"" The computer maps the files of your ownCloud account to the drive letter Z. ""



Though not recommended, you can also mount the ownCloud server using HTTP, leaving the connection unencrypted. If you plan to use HTTP connections on devices while in a public place, we strongly recommend using a VPN tunnel to provide the necessary security.

An alternative command syntax is:

```
net use Z: \\example.com@ssl\owncloud\remote.php\dav /user:youruser  
yourpassword
```

Mapping Drives With Windows Explorer

To map a drive using the Microsoft Windows Explorer:

1. Migrate to your computer in Windows Explorer.
2. Right-click on **[Computer]** entry and select **[Map network drive...]** from the drop-down menu.

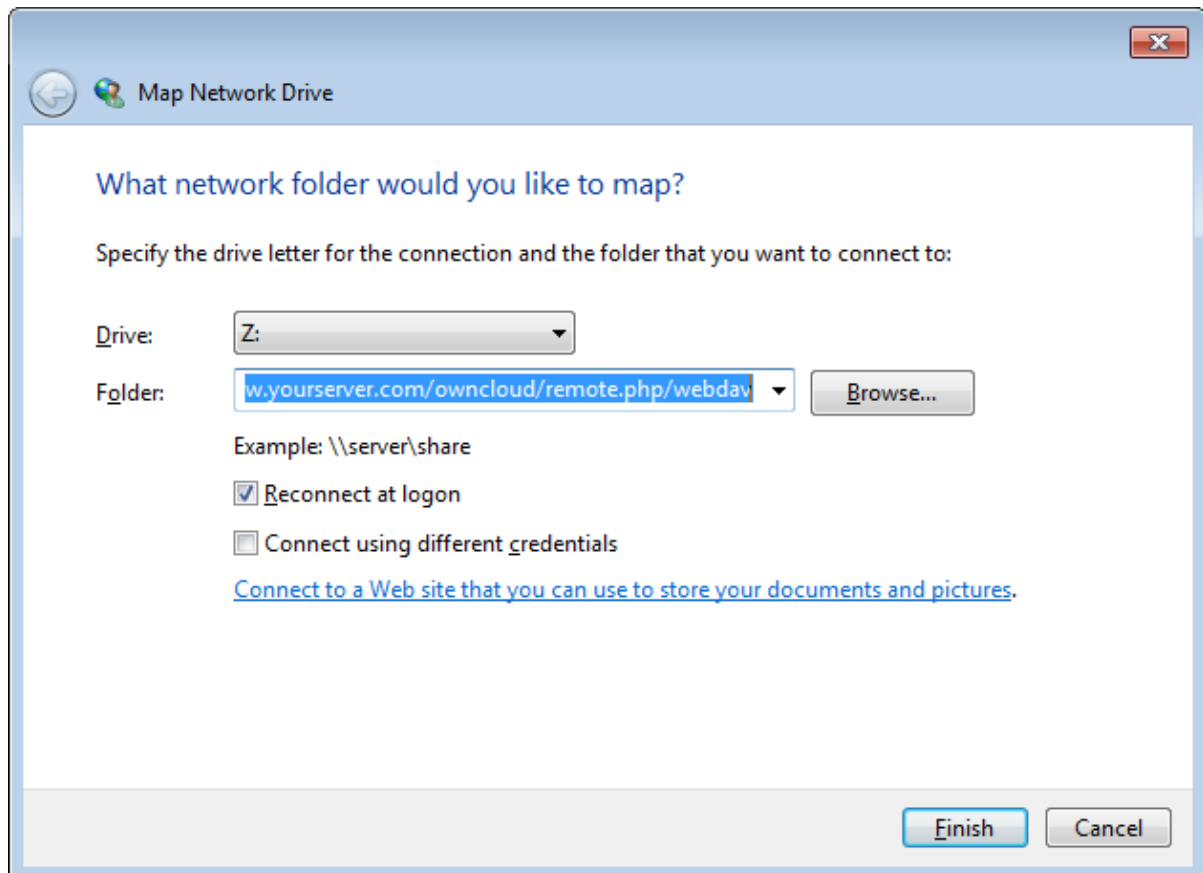
3. Choose a local network drive to which you want to map ownCloud.
4. Specify the address to your ownCloud instance, followed by **/remote.php/webdav**.

For example:

```
https://example.com/owncloud/remote.php/webdav
```



For SSL protected servers, check **Reconnect at logon** to ensure that the mapping is persistent upon subsequent reboots. If you want to connect to the ownCloud server as a different user, check **Connect using different credentials**.



5. Click the **[Finish]** button.
Windows Explorer maps the network drive, making your ownCloud instance available.

Accessing Files Using Cyberduck

Cyberduck is an open source FTP and SFTP, WebDAV, and Amazon S3 browser designed for file transfers on Mac OS X and Windows.



This example uses Cyberduck version 4.2.1.

To use Cyberduck:

1. Specify a server without any leading protocol information. For example:

```
`example.com`
```

2. Specify the appropriate port. The port you choose depends on whether or not your ownCloud server supports SSL. Cyberduck requires that you select a different connection type if you plan to use SSL. For example:

80 (for WebDAV)
443 (for WebDAV (HTTPS/SSL))

3. Use the **More Options** drop-down menu to add the rest of your WebDAV URL into the 'Path' field. For example:

remote.php/webdav

Now Cyberduck enables file access to the ownCloud server.

Accessing public link shares over WebDAV

ownCloud provides the possibility to access public link shares over WebDAV.

To access the public link share, open:

`https://example.com/owncloud/public.php/webdav`

in a WebDAV client, use the share token as username and the (optional) share password as password.



Settings > Administration > Sharing > Allow users on this server to send shares to other servers needs to be enabled in order to make this feature work.

Known Problems

Problem: Windows Does Not Connect Using HTTPS.

Solution 1

The Windows WebDAV Client might not support Server Name Indication (SNI) on encrypted connections. If you encounter an error mounting an SSL-encrypted ownCloud instance, contact your provider about assigning a dedicated IP address for your SSL-based server.

Solution 2

The Windows WebDAV Client might not support TLSv1.1 / TLSv1.2 connections. If you have restricted your server config to only provide TLSv1.1 and above the connection to your server might fail. Please refer to the [WinHTTP](#) documentation for further information.

Problem: The File Size Exceeds the Limit Allowed and Cannot be Saved

You receive the following error message:

Error 0x800700DF: The file size exceeds the limit allowed and cannot be saved.

Solution

Windows limits the maximum size a file transferred from or to a WebDAV share may have. You can increase the value **FileSizeLimitInBytes** in **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\WebClient\Parameters** by clicking on **[Modify]**.

To increase the limit to the maximum value of 4GB, select **Decimal**, enter a value of **4294967295**, and reboot Windows or restart the **WebClient** service.

Problem: Accessing your files from Microsoft Office via WebDAV fails

Solution

Known problems and their solutions are documented in the [KB2123563](#) article.

Problem: WebDAV Drive in Windows Using Self-Signed Certificate

Cannot map ownCloud as a WebDAV drive in Windows using self-signed certificate.

Solution

1. Go to the your ownCloud instance via your favorite Web browser.
2. Click through until you get to the certificate error in the browser status line.
3. View the cert, then from the Details tab, select Copy to File.
4. Save to the desktop with an arbitrary name, for example **myOwnCloud.cer**.
5. Start, Run, MMC.
6. **File > Add/Remove Snap-In.**
7. Select **Certificates > Add > My User Account > Finish > OK.**
8. Dig down to Trust Root Certification Authorities, Certificates.
9. Right-Click **Certificate > Select All Tasks > Import.**
10. Select **[Save Cert]** from the Desktop.
11. Select Place all Certificates in the following Store, click **[Browse]**,
12. Check the Box that says **Show Physical Stores.**
Expand out **Trusted Root Certification Authorities.**
select **Local Computer**, click **[OK]** to complete the Import.
13. Check the list to make sure it shows up.
You will probably need to Refresh before you see it.
Exit MMC.
14. Open Browser, select Tools, Delete Browsing History.
15. Select all but In Private Filtering Data, complete.
16. Go to Internet Options, Content Tab, Clear SSL State.
17. Close browser, then re-open and test.

Problem: Upload Large Files or Upload Takes Long

You cannot download more than 50 MB or upload large Files when the upload takes longer than 30 minutes using Web Client in Windows 7.

Solution

Workarounds are documented in the [KB2668751](#) article.

Problem: The Network Name Cannot be Found

Error 0x80070043 "The network name cannot be found." while adding a network drive.

Solution

Make Windows service **WebClient** start automatically:

1. Open **Control Panel > Administrative Tools > Services**.
2. Find **WebClient** service.
3. Right-click on it and choose **Properties**.
4. Select **Startup type: Automatic**.
5. Click [OK] button.

Or in command prompt (as Admin):

```
sc config "WebClient" start=auto
sc start "WebClient"
```

More details can be found [here](#).

Accessing Files Using cURL

Since WebDAV is an extension of HTTP cURL can be used to script file operations.

To create a folder with the current date as name:

```
curl -u user:pass -X MKCOL
"https://example.com/owncloud/remote.php/dav/files/USERNAME/$(date '+%d-%b-%Y')"
```

To upload a file `error.log` into that directory:

```
curl -u user:pass -T error.log
"https://example.com/owncloud/remote.php/dav/files/USERNAME/$(date '+%d-%b-%Y')/error.log"
```

To move a file:

```
curl -u user:pass -X MOVE --header 'Destination:
https://example.com/owncloud/remote.php/dav/files/USERNAME/target.jpg'
https://example.com/owncloud/remote.php/dav/files/USERNAME/source.jpg
```

To get the properties of files in the root folder:

```
curl -X PROPFIND -H "Depth: 1" -u user:pass
https://example.com/owncloud/remote.php/dav/files/USERNAME/ | xml_pp
<?xml version="1.0" encoding="utf-8"?>
<d:multistatus xmlns:d="DAV:" xmlns:oc="http://owncloud.org/ns" xmlns:s="http://sabredav.org/ns">
  <d:response>
    <d:href>/owncloud/remote.php/webdav/</d:href>
    <d:propstat>
      <d:prop>
        <d:getlastmodified>Tue, 13 Oct 2015 17:07:45 GMT</d:getlastmodified>
        <d:resourcetype>
          <d:collection/>
        </d:resourcetype>
        <d:quota-used-bytes>163</d:quota-used-bytes>
        <d:quota-available-bytes>11802275840</d:quota-available-bytes>
        <d:getetag>"561d3a6139d05"</d:getetag>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>
  <d:response>
    <d:href>/owncloud/remote.php/webdav/welcome.txt</d:href>
    <d:propstat>
      <d:prop>
        <d:getlastmodified>Tue, 13 Oct 2015 17:07:35 GMT</d:getlastmodified>
        <d:getcontentlength>163</d:getcontentlength>
        <d:resourcetype/>
        <d:getetag>"47465fae667b2d0fee154f5e17d1f0f1"</d:getetag>
        <d:getcontenttype>text/plain</d:getcontenttype>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>
</d:multistatus>
```

To get the file id of a file, regardless of location, you need to make a PROPFIND request. This request requires two things:

1. A PROPFIND XML element in the body of the request method.
2. The path to the file that you want to find out more about

Here's an example PROPFIND XML element, which we'll store as `propfind-fileid.xml`.

```
<?xml version="1.0"?>
<a:propfind xmlns:a="DAV:" xmlns:oc="http://owncloud.org/ns">
  <!-- retrieve the file's id -->
  <a:prop><oc:fileid/></a:prop>
</a:propfind>
```



You could pass this directly to the Curl request. However, it can often be easier to create, maintain, and to share, if it's created in a standalone file.

With the file created, make the request by running the following Curl command:

```
curl -u username:password -X PROPFIND \  
-H "Content-Type: text/xml" \  
--data-binary "@propfind-fileid.xml" \  
'http://localhost/remote.php/dav/files/admin/Photos/San%20Francisco.jpg'
```

This will return an XML response payload similar to the following example. It contains the relative path to the file and the fileid of the file.

```
<?xml version="1.0"?>  
<d:multistatus xmlns:d="DAV:" xmlns:s="http://sabredav.org/ns"  
xmlns:cal="urn:ietf:params:xml:ns:caldav" xmlns:cs="http://calendarserver.org/ns/"  
xmlns:card="urn:ietf:params:xml:ns:carddav" xmlns:oc="http://owncloud.org/ns">  
  <d:response>  
    <d:href>/remote.php/dav/files/admin/Photos/San%20Francisco.jpg</d:href>  
    <d:propstat>  
      <d:prop>  
        <oc:fileid>4</oc:fileid>  
      </d:prop>  
      <d:status>HTTP/1.1 200 OK</d:status>  
    </d:propstat>  
  </d:response>  
</d:multistatus>
```



The example above's been formatted for readability, using `xmllint`, which is part of `libxml2`. To format it as it is listed above, pipe the previous command to `xmllint --format -`.

Uploading Files to a Public Link Using cURL

To upload a file "file.txt" to a public link with token "70mX9s7KOZwfmdi" (<https://example.com/s/70mX9s7KOZwfmdi>; no password):

```
curl -k -T file.txt -u "70mX9s7KOZwfmdi:" -H 'X-Requested-With: XMLHttpRequest'  
https://example.com/public.php/webdav/file.txt
```

Sharing Files

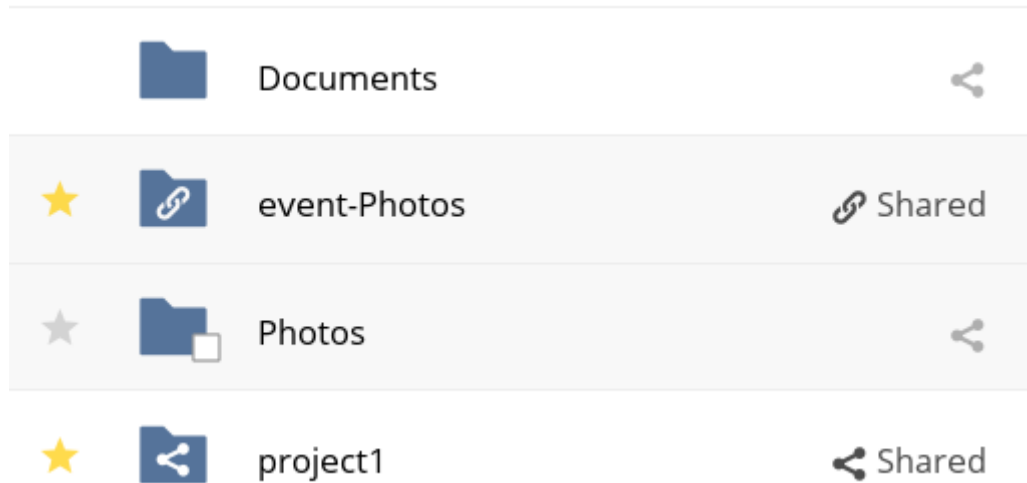
Introduction

Clicking the share icon on any file or folder opens the Details view on the right, where the Share tab has focus.

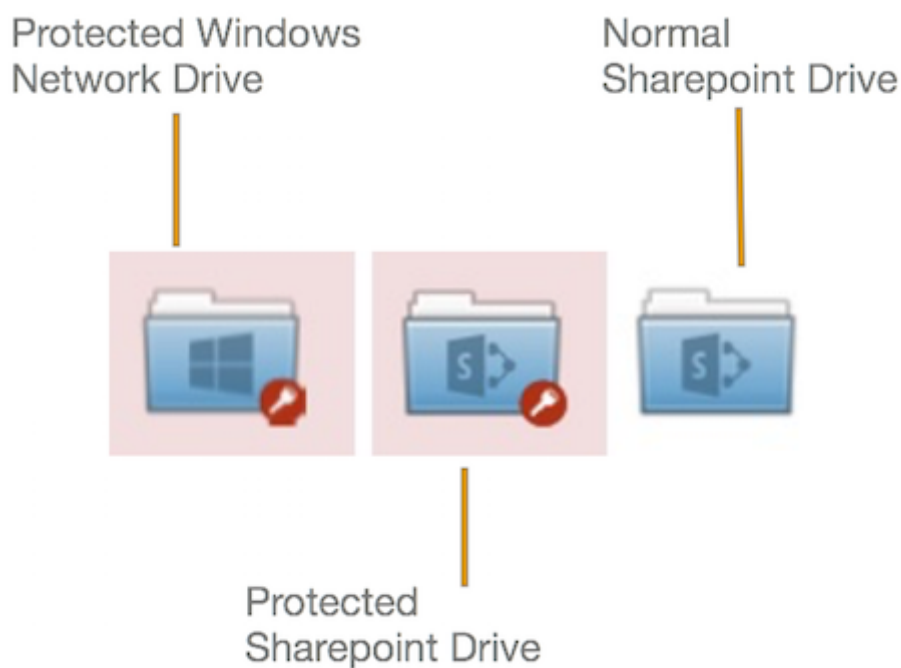
Sharing Status Icons

Any folder that has been shared is marked with the **Shared** overlay icon. Public link

shares are marked with a chain link. Un-shared folders are blank.



If your ownCloud server is the Enterprise edition, you may also have access to Sharepoint and Windows Network Drive file shares. These have special status icons. An icon with a red plugin and background means you have to enter a login to get access to the share.



Creating Shares

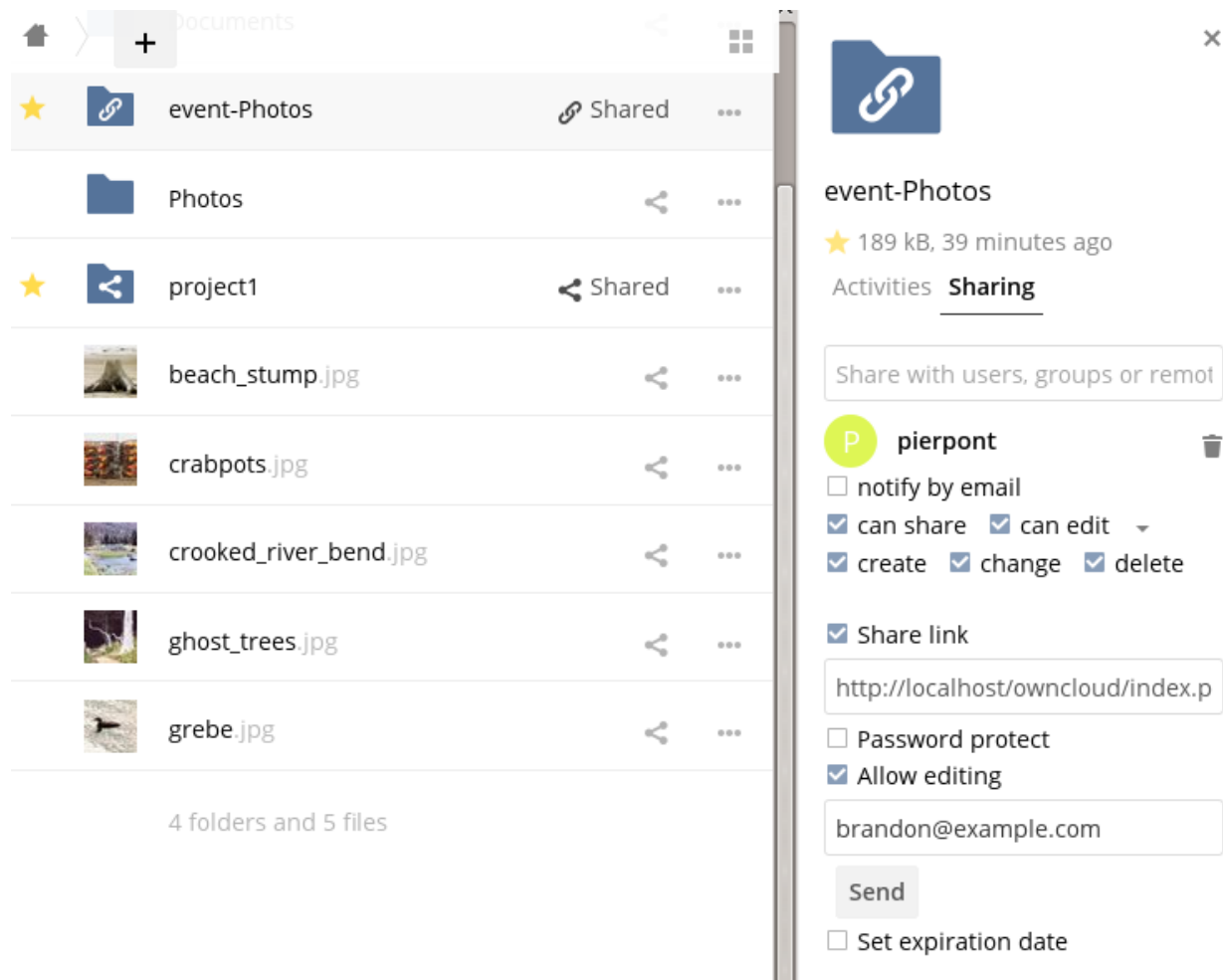
To share a file or folder, immediately below the Share tab, you will see a text field. In this field, you can enter any number of users (whether local to your ownCloud server or remote) or groups who you would like to share the file or folder with.

If username auto-completion is enabled, when you start typing the user or group name ownCloud will automatically complete it for you, if possible.



From 10.0.8, user and group name search results are dependent on a new [configuration setting](#), called `user.search_min_length` (it is set to 4 by default). This setting helps to aid search performance but requires that search terms contain at least the defined number of characters. Consequently, search terms shorter than the defined number of characters will not return a list of matches. However, they will return an existing user or group with a name of the same length as the search term.

After a file or folder has been shared, [Share Permissions](#) can be set on it. In the image below, you can see that the directory "event-Photos" is shared with the user "pierpont", who can share, edit, create, change, and delete the directory.



You can navigate directly to the sharing pane for a file by using the URL: [https://your.owncloud.domain/f/?<\\$fileId>?details=shareTabView](https://your.owncloud.domain/f/?<$fileId>?details=shareTabView), and substituting `<$fileId>` for the file's id.

What Happens When Share Recipients Move Files and Folders?

If a share recipient has "**can edit**" privileges and moves files or folders out of the share, ownCloud stores a backup copy of the moved file/folder in the Deleted Files (Trash) of the share's owner. The user who moved the file/folder out of the share still has the original copy there, along with its attached metadata.

That way, the files/folders are not permanently lost. By clicking the **[Restore]** link, next to the respective file or folder, ownCloud will restore these files/folders to their original location.



Restoring files restores the backup copy for **all users**, including the user that originally moved them, into the original folder.

Sharing Files with Guest Users

Users can also share files and folders with guest users. To do so, your ownCloud administrator will need to have installed the [Guest application](#).

If it's already installed, in the **User and Groups** field of the **Sharing** panel, type the email address of a user who is not already a user in your ownCloud installation. You will then see a popup appear with the suffix (**guest**), as in the screenshot below.

Comments Sharing

User and Groups

Public Links

U

user@example.com (remote)

A

Add user@example.com (guest)

i

After you do that, the content will be shared with the user with all permissions applied, except for the ability to share with other users.

Share with users, groups or remote users...



user@example.com



can share



can edit



Updating Shares

To change any of the properties of a share, again, you first need to view the Share tab. From there, you can:

- Delete a user's access to a share
- Give more users access to a share
- Change a user's share permissions
- Add or remove password protection
- Set or remove a share's expiration date

This functionality is already described in other parts of this documentation and won't be specifically covered here.

Deleting Shares

Despite the name of this section, you don't actually delete a share. Rather what you do is remove the access of user's to whom it's already been shared with. When all users access to a shared resource has been removed, the resource is no longer shared.

To do that, you need to click on the **[rubbish bin]** icon, on the far right-hand side of the name of each user it's been shared with, who should no longer have access to it.

Renaming Shares

Both the sharer *and* all share recipients can rename a share at any time. However, when one user renames a share, it only renames their version; no other users see the new share name. Essentially, the share name remains the same for all other users.

In case that's a little unclear, step through the following scenario:

User Jenny creates a directory called "*Growth Projects 2019*" and shares it with James, Peter, and Sarah. A week later, James renames the share to "*Growth Projects 2019 — Draft!*". James sees the share with the new name, but Jenny, Peter, and Sarah continue seeing the share with its original name ("*Growth Projects 2019*").

This feature may seem a little strange; however, it provides flexibility for all users to manage their files and folders as they see fit.

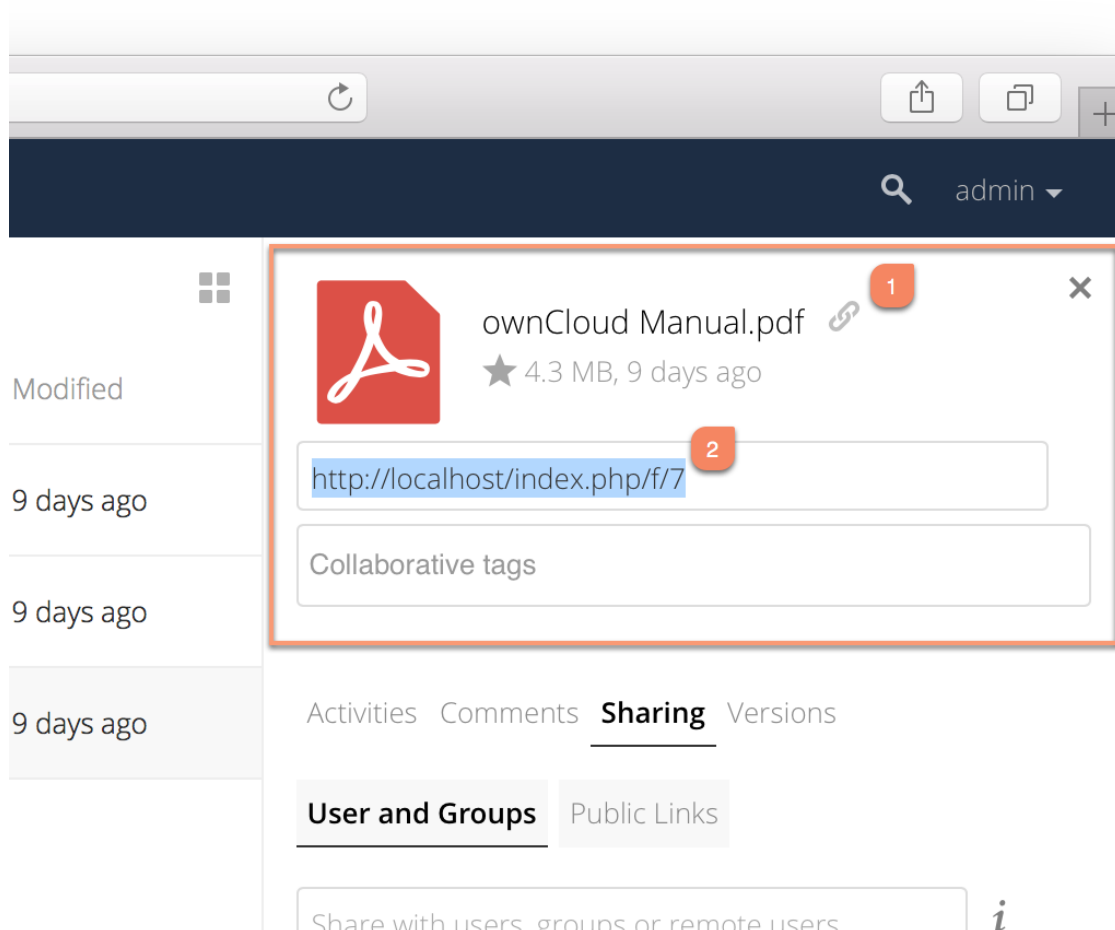
Password Protecting Files

It's also possible to password protect shared files and folders. If you want to do so, then you need to enable this functionality. Specifically, click the checkbox labeled **[Password protect]** under the *"Share Link"* section.


When you do so, you'll see a password field appear. In there, add the password that the user will need to enter to access the shared resource and press the return key.

Using Private Links

Another way to access a file or folder is via a private link. It's a handy way of creating a permanent link for yourself or to point others to a file or folder, within a share, more efficiently. To access the private link, in the Sharing Panel for a file or folder, next to its name you'll see a small link icon (1), as in the screenshot below.



If you click it, a new textbox will appear above the **"Collaborative tags"** field, populated with the link's URI (2).

 Only people who have access to the file or folder can use the link.

Changing The Share Expiration Date

You can set an expiration date on any of user, group and public link shares. The administrator may have set a default expiration for shares. If so, then new shares will have the default expiration. You may adjust or remove the expiration date.

The administrator may have enforced the default expiration to be the maximum expiration. In that case, you must set an expiration date less than or equal to the maximum.

The share will expire at the end of the specified expiration date. Users of the share will

no longer be able to access it.

Creating or Connecting to Federation Share Links

Federated Cloud Sharing allows you to mount file shares from remote ownCloud servers, and manage them just like a local share. In ownCloud 8 the process for creating a new sharing link is easier and more streamlined. See [Using Federation Shares](#) to learn to how to create and connect to new Federated Cloud shares.

Share Permissions

Shares can have a combination of the following five permission types:

Permission	Definition
can share	Allows the users you share with to re-share
can edit	Allows the users you share with to edit your shared files, and to collaborate using the Documents app
create	Allows the users you share with to create new files and add them to the share
change	Allows uploading a new version of a shared file and replacing it
delete	Allows the users you share with to delete shared files

Creating Drop Folders

As of ownCloud version 10.0.2, users can create upload-only, public shares (otherwise known as "Drop Folders"). Drop Folders allow users to upload files to a central location, but don't allow them to either see or change any existing files, which already have been uploaded.

Edit link share: /test

Link Name

test link

☒ Allow editing

☐ Show file listing

Password

Choose a password for the public link

Expiration

Expiration date

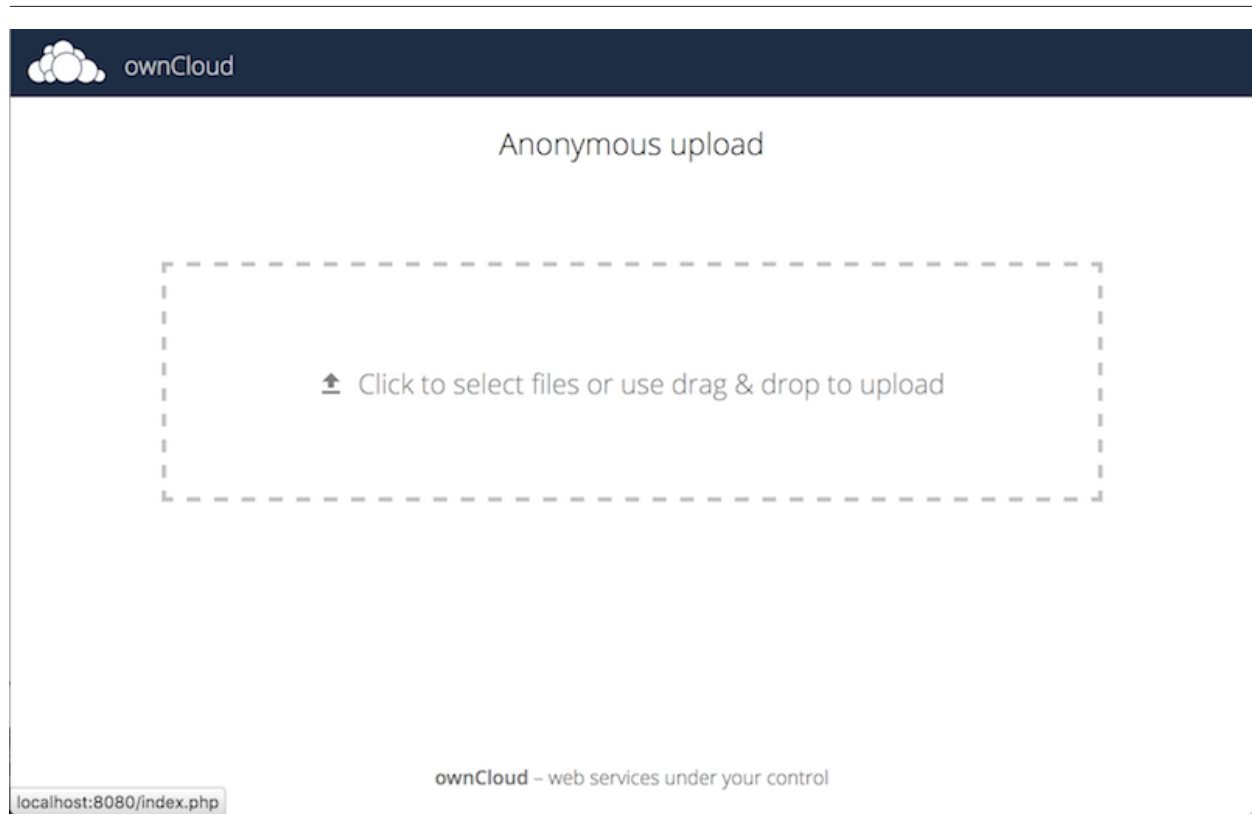
Save Cancel

To create one:

1. View the sharing panel of the folder that you want to share as a Drop Folder, select **Public Links > Create public link**.
2. As with other shares, provide the name in the "**Link Name**" field.
3. Check **[Allow editing]**, un-check **[Show file listing]**, and then un-check **[Allow editing]**.
4. Finally, click **[save]** to complete creation of the share.

Now, as with other public links, you can copy the link to the share and give it out, as and when necessary.

Using Drop Folders

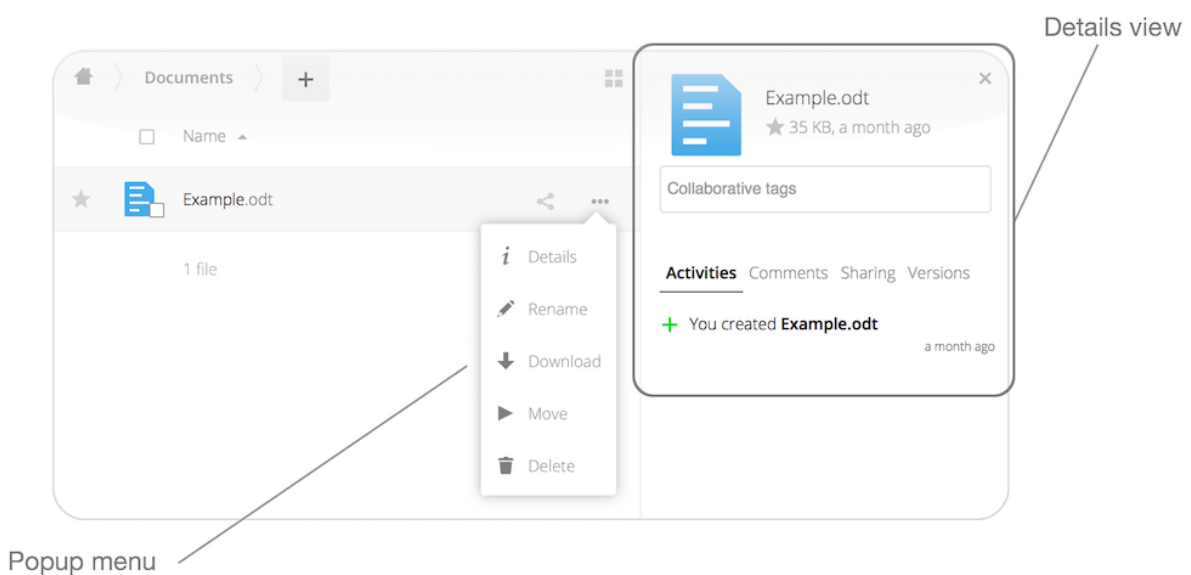


When users open the share link, they will see a page where they can either click to select files to share, or drag-and-drop files directly to share them. After the file's been successfully uploaded, they'll see (where possible) a preview of the file that has been uploaded.

Tagging Files

Introduction

In ownCloud, you can assign one or more tags to files and folders. To do so, go to the **"Details"** view, inside [The Overflow Menu](#).

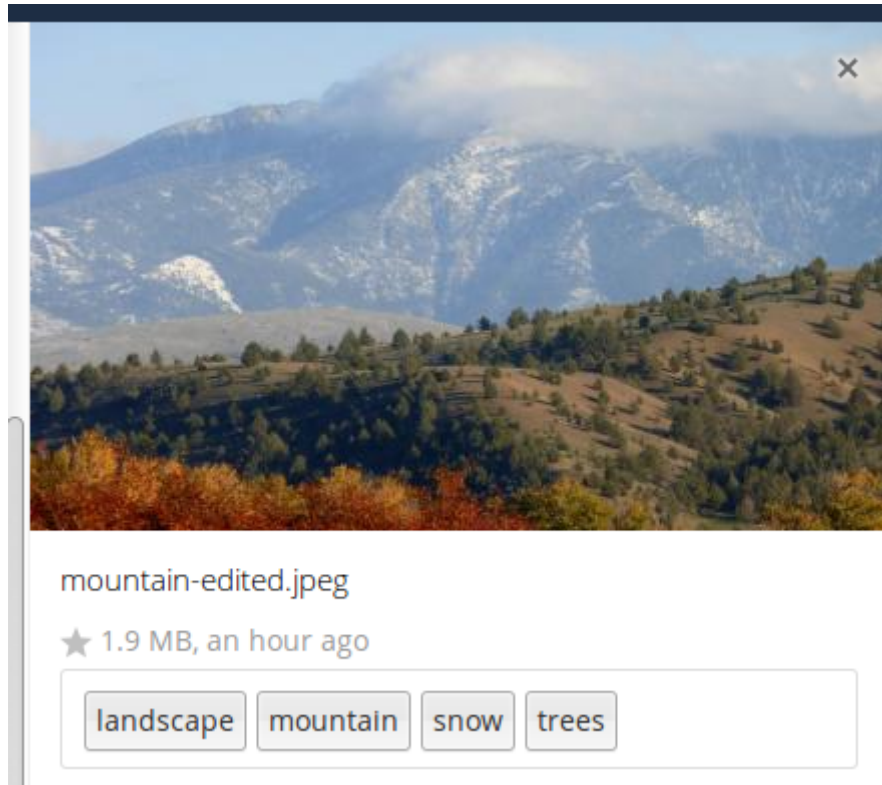


There, you'll see a text field, with the placeholder text **"Collaborative tags"** if no tags have yet been added, below the file's icon, name, and other details.

In that field, type the tag's name. If you want to use multiple words, there is no need to use single or double-quotes. Type as many words as you want for the tag name. When you press the return key your tag will be saved.



All tags are collaborative tags, so they are shared by all users on your ownCloud server.



When you place the cursor inside the tags field, and as you type the tag name, a list of the collaborative tags will appear. If you type a new tag name, the visible tags list will be filtered, based on the text that you've typed.

If you see a tag in the list which is what you had intended to type, or is a better fit than what you had in mind, click on it, and it will be added to the file or folder's tag list. This can save you a lot of time and effort.

Untag a File or Folder

If a file or folder is already tagged, the tag names in the popup list will have a check mark to the left of the tag's name. To remove that tag from the file or folder, click the tag's name. You will see that the check mark disappears.

Edit Tags

To edit a tag, click the pencil icon on the far right-hand side of the tag's name, in the tags popup list. This will display a text box, containing the tag's name. Be sure that you want to change the tag's name, as it will be updated for all users.

Delete Tags

To delete a tag, as above, click the pencil icon on the far right-hand side of the tag's name, in the tags popup list. Next to the text box containing the tag's name, you will also see a delete icon.

Click this to remove the tag from the collaborative tag's list. As with renaming a tag, remember that deleting a tag removes it for all users. So please be sure that you want to do this.

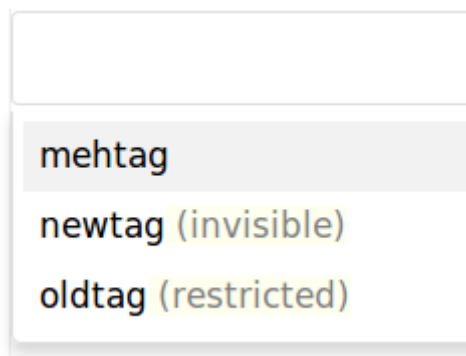
Filter By Tag

To filter by tag, use the **Tags** filter on the left sidebar of the Files page. There are

three types of tags:

Tag	Description
Visible	All users may see, rename, and apply these tags to files and folders
Restricted	Tags are assignable and editable only to the users and groups which have permission to use them. Other users can filter files by restricted tags, but cannot tag files with them or rename them. The tags are marked (restricted)
Invisible	Visible only to ownCloud admins

When you use the **Tag** filter on your Files page you'll see something like the following image. If you do not have Admin rights then you will not see any invisible tags.



Encrypting Your ownCloud Files

Introduction

ownCloud includes an Encryption app, and when it is enabled by your ownCloud administrator all of your ownCloud data files are automatically encrypted. Encryption is server-wide, so when it is enabled you cannot choose to keep your files unencrypted. You don't have to do anything special, as it uses your ownCloud login as the password for your unique private encryption key. Just log in and out and manage and share your files as you normally do, and you can still change your password whenever you want.

Its main purpose is to encrypt files on remote storage services that are connected to your ownCloud server, such as Dropbox and Google Drive. This is an easy and seamless way to protect your files on remote storage. You can share your remote files through ownCloud in the usual way, however you cannot share your encrypted files directly from Dropbox, Google Drive, or whatever remote service you are using, because the encryption keys are stored on your ownCloud server, and are never exposed to outside service providers.

If your ownCloud server is not connected to any remote storage services, then it is better to use some other form of encryption such as file-level or whole disk encryption. Because the keys are kept on your ownCloud server, it is possible for your ownCloud admin to snoop in your files, and if the server is compromised the intruder may get access to your files. (Read [How ownCloud uses encryption to protect your data](#) to learn more.)

Using Encryption

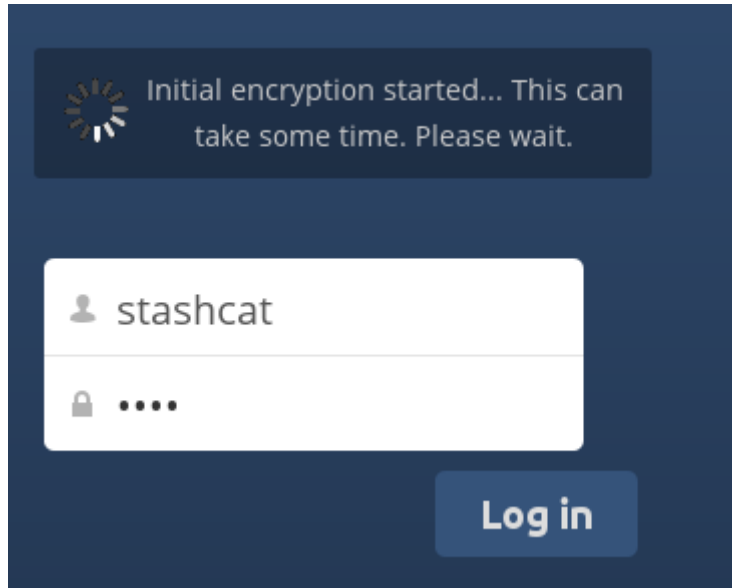
ownCloud encryption is pretty much set it and forget it, but you have a few options you can use.

When your ownCloud admin enables encryption for the first time, you must log out and then log back in to create your encryption keys and encrypt your files. When

encryption has been enabled on your ownCloud server you will see a yellow banner on your Files page warning you to log out and then log back in.

Encryption App is enabled but your keys are not initialized, please log-out and log-in again

When you log back in it takes a few minutes to work, depending on how many files you have, and then you are returned to your default ownCloud page.



You must never lose your ownCloud password, because you will lose access to your files. However, there is an optional recovery option that your ownCloud administrator may enable; see the Recovery Key Password section (below) to learn about this.

Which Files Are Never Encrypted

Only the **data** in the files in `data/<user>/files`, and external storages (*if enabled*), is encrypted, *not* the filenames or folder structures. The following files are **never** encrypted:

- Existing files in the trash bin & Versions. Only new and changed files after encryption is enabled are encrypted.



You can post encrypt existing files via an [occ encryption command](#).

- Existing files in Version
- Image thumbnails
- Previews from the Files app.
- The search index from the full text search app.
- Third-party app data

There may be other files that are not encrypted.



Only files that are exposed to third-party storage providers are guaranteed to be encrypted.

Sharing Encrypted Files

Only users who have private encryption keys have access to shared encrypted files and folders. Users who have not yet initialized their private encryption keys will not have access to encrypted shared files; they will see folders and filenames, but will not be able to open or download the files. They will see a yellow warning banner that says:

```
"" Encryption App is enabled but your keys are not initialized, please log-out and log-in again. ` ""
```

Share owners may need to re-share files after encryption is enabled; users trying to access the share will see a message advising them to ask the share owner to re-share the file with them. For individual shares, un-share and re-share the file. For group shares, share with any individuals who can't access the share. This updates the encryption, and then the share owner can remove the individual shares.

Recovery Key Password

If your ownCloud administrator has enabled the recovery key feature, you can choose to use this feature for your account. If you enable "Password recovery" the administrator can read your data with a special password. This feature enables the administrator to recover your files in the event you lose your ownCloud password. If the recovery key is not enabled, then there is no way to restore your files if you lose your login password.

Encryption

Enable recovery key (allow to recover users files in case of password loss):

<input type="password" value="....."/>	Recovery key password
<input type="password" value="....."/>	Repeat Recovery key password
<input checked="" type="radio"/> Enabled	
<input type="radio"/> Disabled	

Change Private Key Password

This option is only available if your log-in password, but not your encryption password, was changed by your administrator. This can occur if your ownCloud provider uses a external user back-end (for example, LDAP) and changed your login password using that back-end configuration. In this case, you can set your encryption password to your new login password by providing your old and new login password. The Encryption app works only if your login password and your encryption password are identical.

Managing Deleted Files

Introduction

When you delete a file in ownCloud, it is not immediately deleted permanently. Instead, it is moved into the trash bin. It is not permanently deleted until you manually delete it, or when the Deleted Files app deletes it to make room for new files.

Find your deleted files by clicking on the **[Deleted files]** button on the Files page of the ownCloud Web interface. You'll have options to either restore or permanently delete files.

Quotas

Deleted files are not counted against your storage quota. Only files that originate with users count against their quotas, not files shared with them that originate from other users. (See [webgui/quota](#) to learn more about quotas.)

What Happens When Shared Files Are Deleted

Deleting files gets a little complicated when they are shared files, as this scenario illustrates:

1. User1 shares a folder "test" with User2 and User3
2. User2 (the recipient) deletes a file/folder "sub" inside of "test"
3. The folder "sub" will be moved to the trashbin of both User1 (owner) and User2 (recipient)
4. But User3 will not have a copy of "sub" in her trash bin

When User1 deletes "sub" then it is moved to User1's trash bin. It is deleted from User2 and User3, but not placed in their trash bins.

When you share files, other users may copy, rename, move, and share them with other people, just as they can for any computer files; ownCloud does not have magic powers to prevent this.

How the Deleted Files app Manages Storage Space

To ensure that users do not run over their storage quotas, the Deleted Files app allocates a maximum of 50% of their currently available storage quota to deleted files. If your deleted files exceed this limit, ownCloud deletes the oldest files (files with the oldest timestamps from when they were deleted) until it meets the memory usage limit again.

ownCloud checks the age of deleted files every time new files are added to the deleted files. By default, deleted files stay in the trash bin for 180 days. The ownCloud server administrator can adjust this value in the [config.php](#) file by setting the [trashbin_retention_obligation](#) value. Files older than the [trashbin_retention_obligation](#) value will be deleted permanently. Additionally, ownCloud calculates the maximum available space every time a new file is added. If the deleted files exceed the new maximum allowed space ownCloud will expire old deleted files until the limit is met once again.

Large File Uploads

When uploading files through the web client, ownCloud is limited by PHP and Apache configurations. By default, PHP is configured for only 2 megabyte uploads. As this default upload limit is not entirely useful, we recommend that your ownCloud admin increase the ownCloud variables to sizes appropriate for users.

Modifying certain ownCloud variables requires administrative access. If you require larger upload limits than have been provided by the default (or already set by your administrator):

- Contact your administrator to request an increase in these variables
- Refer to the section in the [Admin Documentation](#) that describes how to manage file upload size limits.

File Lifecycle Management

Introduction

With File Lifecycle Management, ownCloud provides a toolset for administrators to automatically move user files into a dedicated archive a certain time after they were uploaded. Unless they are restored, archived files will be permanently deleted once they exceed a certain time in the archive. The following sections provide guidance for users on ownCloud installations that have File Lifecycle Management policies enabled.



Only files will be archived, folder structures are kept available.

Archiving Process

To see when and if files are scheduled for archiving, a user may select a file within the files list to open the detail view in the sidebar. In the sidebar, they can see the remaining days until it will be automatically archived. Archiving usually takes place once a day. When a file is due to be archived, it will show 'Scheduled for archive today!'.



Portugal.jpg [link](#)

★ 238 KB, a month ago

Scheduled for archive in 7 days

Collaborative tags

Activities Comments Sharing Versions

+ You created Portugal.jpg

2 minutes ago

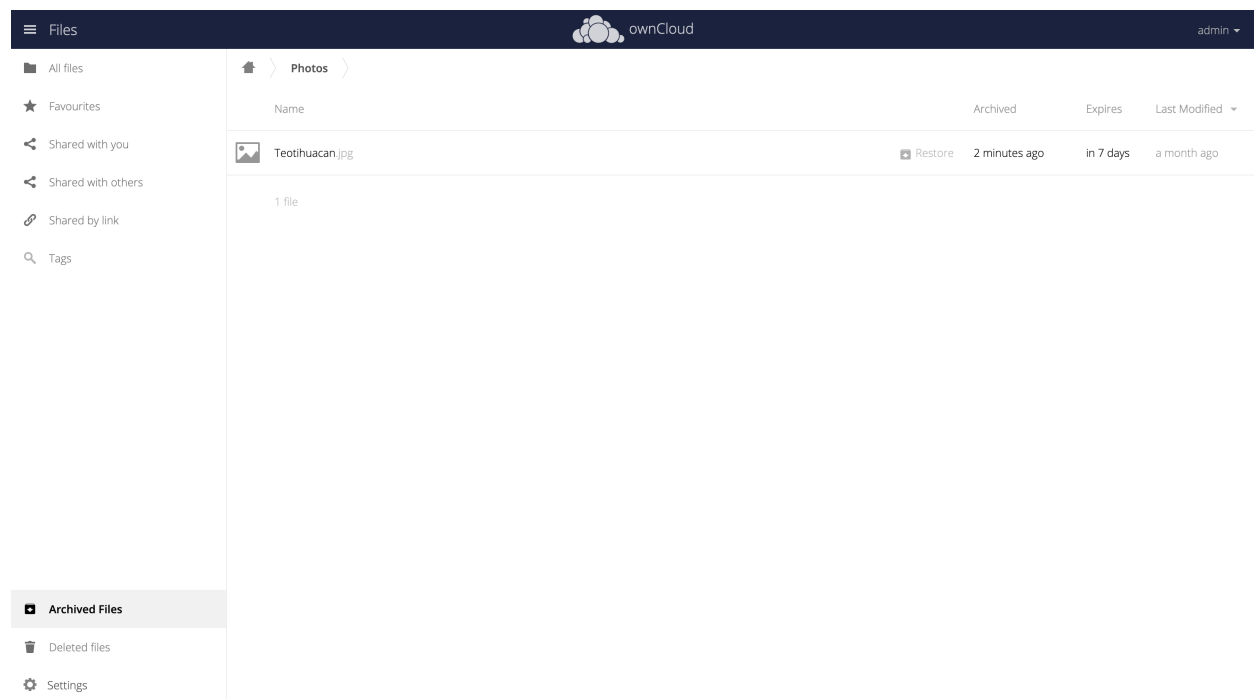
Depending on the policy configuration, users may be able to archive their own files using a file action in the file list.

Whilst a file is in the archive, it cannot be used in the Web interface or with the ownCloud Clients, but can be located using the archive browser. Existing metadata including shares, comments and tags are preserved, but will not be available for archived files.

Browsing the Archive

Users can browse the archive in a similar fashion to the 'Deleted Files', using the 'Archived Files' file list available on the bottom left of the files view. Folder structures

are recreated showing the paths that were present at the time a file was archived.



Restoring Files

Depending on the policy configuration, users may be able to restore files on their own in 'Archived Files' by clicking the 'Restore' action on the file or folder row. Specific policies may require different permissions to access this option, or permanently disable it. Restored files can be used for the same amount of time as if they were uploaded before they will be archived again.

Activities

File Lifecycle Management events (like archiving/restoring a file) are added to the Activity history. These can be viewed within the 'Activity' tab in the sidebar or within the Activity Stream. Within the personal settings page, users can choose to receive e-mails related to these events.



Teotihuacan.jpg [🔗](#)

★ 223 KB, a month ago

Scheduled for archive in a day

Collaborative tags

Activities Comments Sharing Versions

- ▼ Teotihuacan.jpg was restored 7 days ago
- ▼ Teotihuacan.jpg was moved to the archive 7 days ago
- + You created Teotihuacan.jpg 13 days ago

Public Link Shares

With ownCloud X (10.0), we introduced the ability to create multiple public links per file or folder. This offers a lot of flexibility for creating different kinds of share links for a single file or folder, such as *different passwords*, *expiry dates*, and *permissions*.

As of ownCloud version 10.0.2 you can [create Drop Folders](#), where users can upload files to a central location, but not be able to change any existing ones, nor see other files which already have been uploaded.

Creating Public Link Shares

To create a public link share, first view the Sharing Panel of the file or folder that you want to create a public link share for. Then, click the **[Public Links]** button, and then click **[Create public link]**. After you do, the public link share dialog will appear, which you can see below.



Photos



★ 663 KB, 2 months ago

Collaborative tags

Activities Comments Sharing

User and Groups

Public Links

There are currently no link shares, you can create one

Create public link

Anyone with the link has access to the file/folder

As with other shares, provide the name in the "**Link Name**" field, and fill out the options that suit what you want the link to support. You can find details of what each option does below.

×

Edit link share: /Photos

Link name

Public link

☒ Download / View

Recipients can view or download contents.

☐ Download / View / Edit

Recipients can view, download and edit contents.

☐ Download / View / Upload

Recipients can view, download and upload contents.

☐ Upload only (File Drop)

Receive files from multiple recipients without revealing the contents of the folder.

Password

Expiration

Choose an expiration date

Cancel

Remove password

Save

Finally, click the **[save]** button to complete creation of the share. Now that the share is created, you can:

- Copy the link to the share and give it out
- Update the share’s settings
- Share the link via social media and email
- Delete the public link

Share Settings Explained

Setting	Description
Download / View	Allows recipients to view or download the public link’s contents.
Download / View / Edit	Allows recipients to view, download, edit, and delete the public link’s contents.
Download / View / Upload	Allows recipients to view, download and upload the public link’s contents.

Setting	Description
Upload only (File Drop)	Allows users to create a drop folder , which can receive files from multiple recipients without revealing the contents of the folder.
Password	Sets a password for protecting the link.
Expiration	Sets an expiry date for the link.

Using Federation Shares

Introduction

Federation Sharing allows you to mount file shares from remote ownCloud servers, in effect creating your own cloud of ownClouds. You can create direct share links with users on other ownCloud servers.



Sharing to groups from federated ownCloud instances is not supported.

How Federated Sharing Works

Federated sharing, conceptually, is not that sophisticated a concept. Here's how it works.

Say we have three users: *James*, *Mary*, and *Paul*. James has a folder (Majorca-Holiday-Pics) which he shares with Mary, who's on a separate ownCloud instance. Mary, in turn, (re)shares the folder with Paul, who's on the same ownCloud instance as Mary.

You might think that there are two — even three — copies of the shared folder. In reality, there's only one. *In effect*, there are three — all owned by the original sharer (James).

The key point to keep in mind is that when a share is re-shared, it's shared, internally, on behalf of the original owner. To keep track of all this, during the share process references are created between the shares, that show:

- James was the original owner of the shared resource
- James shared a copy with Mary and Mary re-shared a copy with Paul.

Creating a New Federation Share

Federation sharing is enabled on new or upgraded ownCloud installations by default. Follow these steps to create a new share with other ownCloud 9 servers:

1. Go to your **Files** page and click the **[share]** icon on the file or directory you want to share. In the sidebar enter the username and URL of the remote user in this form: `<username>@<oc-server-url>`. In this example, that is `layla@remote-server/owncloud`. The form automatically echoes the address that you type and labels it as "remote". Click on the label.

Comments **Sharing** Versions



☐ Share link

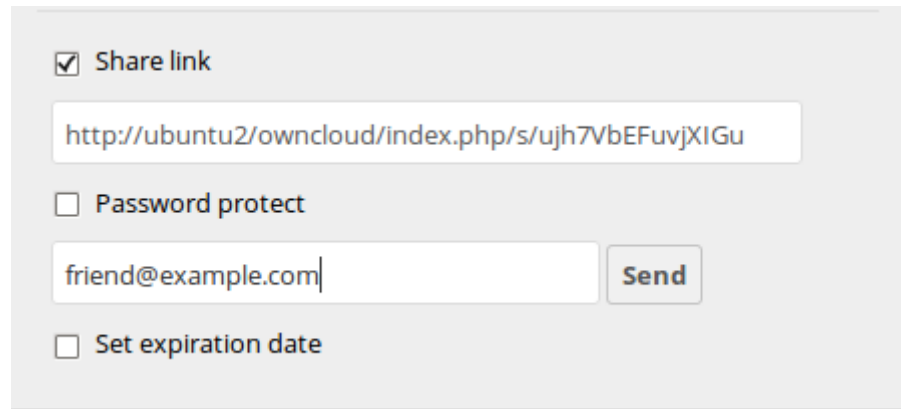
-
2. When your local ownCloud server makes a successful connection with the remote ownCloud server you'll see a confirmation. Your only share option is **Can edit**.

Click the Share button anytime to see who you have shared your file with. Remove your linked share anytime by clicking the trash can icon. This only unlinks the share, and does not delete any files.

Creating a New Federated Cloud Share via Email

Use this method when you are sharing with users on ownCloud 8.x and older.

What if you do not know the username or URL? Then you can have ownCloud create the link for you and email it to your recipient.



A screenshot of the 'Share link' form in ownCloud. The form is light gray and contains the following elements: a checked checkbox labeled 'Share link', a text input field containing the URL 'http://ubuntu2/owncloud/index.php/s/ujh7VbEFuvjXIGu', an unchecked checkbox labeled 'Password protect', a text input field containing the email 'friend@example.com' next to a 'Send' button, and an unchecked checkbox labeled 'Set expiration date'.

When your recipient receives your email they will have to take a number of steps to complete the share link. First they must open the link you sent them in a Web browser, and then click the [Add to your ownCloud] button.



The **Add to your ownCloud** button changes to a form field, and your recipient needs to enter the URL of their ownCloud server in this field and press the return key, or click the arrow.



Next, they will see a dialog asking to confirm. All they have to do is click the btn[Add remote share] button and they’re finished.

Remove your linked share anytime by clicking the [trash can] icon. This only unlinks the share, and does not delete any files.

Session Management



Introduction

The personal settings page allows you to have an overview of the connected browsers and clients. It is accessed by selecting the **Settings > Personal > Security**.

Sessions

Sessions

These are the web, desktop and mobile clients currently logged in to your ownCloud.

Browser	Most recent activity	
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (K...	2 minutes ago	
Mozilla/5.0 (iOS) ownCloud-iOS/3.8.0	4 minutes ago	
Mozilla/5.0 (Windows) mirall/2.5.2 (build 11181)	3 minutes ago	

The sessions list shows which browsers and clients were recently, and are actively connected to your ownCloud installation. You can use the trash icon, at the far right-hand side of any session, to terminate it.

Please Be Aware of Two Things

1. If you want to lock out a user, you need to change their password **before** you terminate their session.
2. The currently logged-in user cannot terminate their own session from the sessions list.

App Passwords

App passwords / tokens

You've linked these apps.

Name	Most recent activity	
Client	seconds ago	

App passwords or tokens are passcodes that give an app or device permissions to access your ownCloud account. Use them as a security measure to hide your actual password which you may only want to use for web interface login. Use the credentials below to configure your app or device.

Username	<input type="text" value="admin"/>
Password / Token	<input type="text" value="SBILI-VXEEM-JXCXX-QBSJJ"/> <input type="button" value="Done"/>

Underneath the "App passwords / tokens" list is a button to create a new app or device-specific username and password. The username will be pre-filled to the currently logged in user, and a random password will be generated. You can change both of these, if so desired. When you're happy with the username and password, click the **[done]** button. You can use the trash icon, at the far right-hand side of any password, to delete it





Once the password / token is created, **it cannot be changed**; it can only be deleted.



We recommend that you generate tokens for every device you want to connect to your ownCloud instance, as this will allow you to disconnect connections individually, if necessary.



Version Control



ownCloud supports simple version control system for files. Versioning creates backups of files which are accessible via the Versions tab on the Details sidebar. This tab contains the history of the file where you can roll back a file to any previous version. Changes made at intervals greater than two minutes are saved in data/[user]/versions.



 lesson-2.html 

★ 10 kB, a minute ago

Activities Sharing **Versions**

 ↓ 18 minutes ago 

 ↓ 19 minutes ago 

 ↓ 19 minutes ago 



You can navigate directly to the versions pane for a file by using the URL: [https://your.owncloud.domain/f/?<\\$fileId>?details=versionsTabView](https://your.owncloud.domain/f/?<$fileId>?details=versionsTabView), and substituting **<\$fileId>** for the file's id.

To restore a specific version of a file, click the [circular arrow] to the left. Click on the [timestamp] to download it.

The versioning app expires old versions automatically to make sure that the user doesn't run out of space. This pattern is used to delete old versions:

- For the first second we keep one version
- For the first 10 seconds ownCloud keeps one version every 2 seconds
- For the first minute ownCloud keeps one version every 10 seconds
- For the first hour ownCloud keeps one version every minute
- For the first 24 hours ownCloud keeps one version every hour
- For the first 30 days ownCloud keeps one version every day
- After the first 30 days ownCloud keeps one version every week

The versions are adjusted along this pattern every time a new version gets created.

The version app never uses more than 50% of the user's currently available free space. If the stored versions exceed this limit, ownCloud deletes the oldest versions until it meets the disk space limit again.

Online Collaboration

Collabora Online

Collabora Online is a powerful LibreOffice-based online office that supports all major document, spreadsheet and presentation file formats, and is integrable with ownCloud.

Secure View

Secure View is a way to securely distribute information, without that information being able to leave your organisation. It provides the ability to restrict access to documents, in *read-only* shares, so that recipients cannot copy, download, or edit shared files.

When enabled, documents will be viewed through Collabora Online with a watermark. In addition, the ability to print and export files (with watermarks) is available, once Secure View is enabled.



This feature is **not** available in the community edition, or for Public Links.

How to Use Secure View

Comments Sharing

User and Groups

Public Links

Share with users or groups...



editor



can share



can edit



Secure View (with watermarks)



can print / export PDF

To enable Secure View for an existing, or new, share:

- enable [**Secure View (with watermarks)**]
- enable [**can print / export PDF**] (*optional*). This allows documents to be printed and exported through Collabora Online.



"*can edit*" and "*Secure View (with watermarks)*" are mutually exclusive. If "*can edit*" is enabled and a user enables "*Secure View (with watermarks)*", then "*can edit*" is automatically disabled.

When "*Secure View (with watermarks)*" is enabled, any attempts to download the file will be blocked, as exemplified in the screenshot below. Additionally, select, copy, and paste are disabled.



Storage Quotas

Introduction

Your ownCloud admin has the option to set a storage quota on users. Look at the top of your Personal page to see what your quota is, and how much you have used.



You have used **70.6 MB** of the available **5 GB**

It may be helpful to understand how your quota is calculated. Metadata (*thumbnails, temporary files, cache, and encryption keys*) takes up about 10% of disk space, but is not counted against user quotas. Some apps store information in the database, such as the Calendar and Contacts apps. This data is excluded from your quota.

When other users share files with you, the shared files count against the original share owner's quota. When you share a folder and allow other users or groups to upload files to it, all uploaded and edited files count against your quota. When you re-share files shared with you, the re-share still counts against the quota of the original share owner.

Encrypted files are a little larger than unencrypted files; the unencrypted size is calculated against your quota. Deleted files that are still in the trash bin do not count against quotas. The trash bin is set at 50% of quota. Deleted file aging is set at 30 days. When deleted files exceed 50% of quota then the oldest files are removed until the total is below 50%.

When version control is enabled, the older file versions are not counted against quotas. If you create a public link share via URL, and allow uploads, any uploaded files count against your quota.

Quotas Are Neither Hard Nor Soft

Quotas are not strictly hard quotas, but nor are they soft quotas either. They're more hard quotas with the exception of always being able to restore deleted files.

Why? Well, say that a user's Trash bin contains one or more files. The user uploads several files and in the process reaches their quota limit.

If hard quotas were enforced, the user would be prevented from being able to restore any deleted file. However, with the Trash bin exception in place, files can always be restored, but no new files can be uploaded, once a quota has been reached.

This user workflow may sound peculiar, but not allowing a user to restore files would make for a poor user experience. Given that, this exception is allowed.

Configuring External Storage

The External Storage application allows you to mount external storage services, such as Google Drive, Dropbox, Amazon S3, SMB/CIFS filesystems, and FTP servers in ownCloud. Your ownCloud server administrator controls which of these are available to you. Please see [Configuring External Storage \(GUI\)](#) in the ownCloud Administrator's manual for configuration howtos and examples.

Connecting to SharePoint (Enterprise only)

Introduction

Native SharePoint support has been added to ownCloud Enterprise Subscription as a secondary storage location for SharePoint 2007, 2010 and 2013. To the user, these appear as normal ownCloud mounts, with bi-directional updates in any ownCloud client: desktop, mobile, or Web.

There is one difference, and that is ownCloud sharing is intentionally disabled for SharePoint mountpoints in order to preserve SharePoint access controls, and to ensure that content is properly accessed as per SharePoint rules. Your ownCloud admin may optionally allow users to mount their own SharePoint libraries.

Accessing SharePoint Folders

When you first log in to ownCloud, the Web interface shows a gray bar behind all SharePoint folders. The gray bar disappears when the mountpoint is verified by the server. If you see a red error bar, you'll see either an hourglass that indicates a connection error, or a key to indicate that authentication is required.

Your ownCloud admin has the option to configure SharePoint credentials so that you are authenticated automatically, or you may be required to enter your credentials. If you have to enter your credentials, click the **[red bar]** and you'll get a login window. You should only have to do this once, as ownCloud will store your credentials.

If your SharePoint login ever changes, go to your Personal page to update it in the [Sharepoint Personal Configuration](#) section.

Personal Page

You can manage your SharePoint connections in the [Sharepoint Personal Configuration](#) section of your ownCloud Personal page. You'll see two sections: the **Admin added mount points** section lists SharePoint mounts controlled by your ownCloud admin. If users have permissions to mount their own SharePoint libraries you'll also see a **Personal mount points** section.

There are two types of authentication available to you. If you have multiple SharePoint libraries that use the same authentication, enter your credentials in **Sharepoint Personal Configuration**. Then follow these steps to add your libraries:

- Enter the name of your local mountpoint in the **Local Folder Name** column.
- Enter your SharePoint server URL.
- Click the little refresh icon to the left of the **Document Library** field. If your credentials and URL are correct you'll get a dropdown list of SharePoint libraries to choose from.
- Select the document library you want to mount.
- Select "Use user credentials".
- Click the **[save]** button, and you're done

You may elect to use different authentication credentials for some of your SharePoint libraries. For these, you must first select **use custom credentials**, and then fill in the mountpoint and SharePoint site URL. Then ownCloud can authenticate you, and you can click the **[refresh]** icon to see your libraries. Then select the library you want to mount and click the **[save]** button.

Desktop and Mobile Synchronization

For synchronizing files with your desktop computer, we recommend using the **ownCloud Sync Client** for Windows, Mac OS X and Linux.

The ownCloud Desktop Sync Client enables you to connect to your private ownCloud Server. You can create folders in your home directory, and keep the contents of those folders synced with your ownCloud server. Simply copy a file into the directory and the ownCloud desktop client does the rest. Make a change to the files on one computer, it will flow across the others using these desktop sync clients. You will always have your latest files with you wherever you are.

Its usage is documented separately in the **ownCloud Desktop Client Manual**.

Mobile Clients

Visit your Personal page in your ownCloud Web interface to find download links for Android and iOS mobile sync clients. Or, visit the **ownCloud download page**.

Visit the **ownCloud documentation page** to read the mobile apps user manuals.

Apps

In this section you will find all the details you need on some of the core apps available with ownCloud.

The Activity App

Introduction

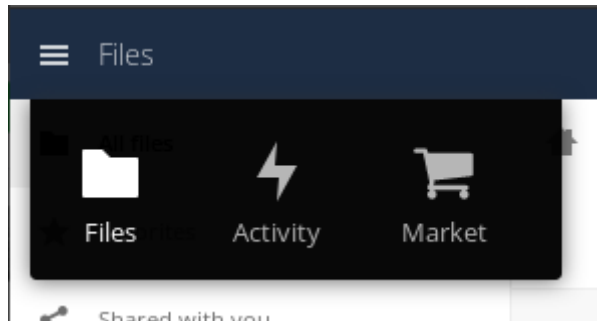
The ownCloud Activity app gathers all your file or folder related actions in one place for you to review and can notify you about them via email as well. You can decide, in detail, which file and folder actions are listed in the Activity stream, and for which file or folder actions to receive email notifications. By using the app, you can ensure that you never miss an important event related to content in ownCloud and always be up-to-date on all activities of your files and folders.



The Activity App is shipped and enabled by default. If it is not enabled, contact your ownCloud administrator to have it enabled.

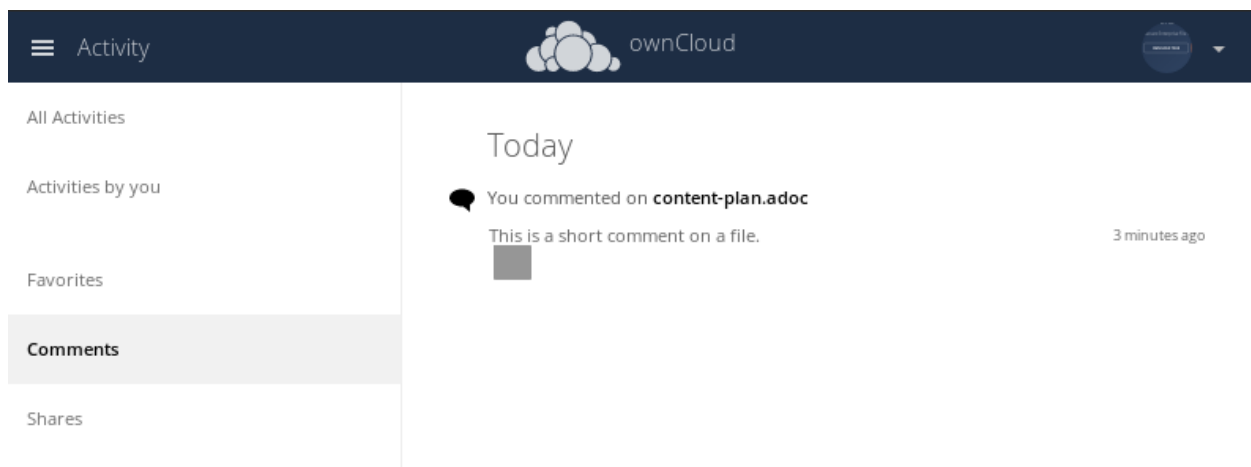
Viewing Notifications

To view notifications, click the hamburger menu, in the top left-hand corner of the WebUI, and then click "Activity", as you can see in the screenshot below.



By default, you see all activity related to your files and folders. However, by using the left-hand navigation menu, you can filter activity by:

- Activities by you
- Activities by others
- Favorites
- Comments
- Shares



Configuring the Activity App

To configure your Activity preferences navigate to **Settings > Personal > General > Activity**. You can configure notifications for when:

- Files and folders are created, changed, deleted, restored (from the trash bin), and shared.
- Files and folders are shared from another server.
- A publicly shared file or folder was downloaded.
- Comments are added to a file.
- System tags for a file have been modified.

To do so, enable or disable the relevant checkboxes in the Activity settings panel, as in the screenshot below.

Activity

Mail	Stream	
<input type="checkbox"/>	<input checked="" type="checkbox"/>	A new file or folder has been created
<input type="checkbox"/>	<input checked="" type="checkbox"/>	A file or folder has been changed
<input type="checkbox"/>	<input type="checkbox"/>	Limit notifications about creation and changes to your favorite files (Stream only)
<input type="checkbox"/>	<input checked="" type="checkbox"/>	A file or folder has been deleted
<input type="checkbox"/>	<input checked="" type="checkbox"/>	A file or folder has been restored
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	A file or folder has been shared
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	A file or folder was shared from another server
<input type="checkbox"/>	<input checked="" type="checkbox"/>	A public shared file or folder was downloaded
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Comments for files (always listed in stream)
<input type="checkbox"/>	<input checked="" type="checkbox"/>	System tags for a file have been modified

- ☒ List your own file actions in the stream
- ☐ Notify about your own actions via email

Send emails: Hourly ▾



Sharing actions are only visible to the sharer and recipient.

Limiting Notifications

For users with lots of activity, it is possible to limit the Activity stream to 'Favorites' to avoid noise.

Activity

Mail	Stream	
<input type="checkbox"/>	<input checked="" type="checkbox"/>	A new file or folder has been created
<input type="checkbox"/>	<input checked="" type="checkbox"/>	A file or folder has been changed
<input type="checkbox"/>	<input type="checkbox"/>	Limit notifications about creation and changes to your favorite files (Stream only)
<input type="checkbox"/>	<input checked="" type="checkbox"/>	A file or folder has been deleted
<input type="checkbox"/>	<input checked="" type="checkbox"/>	A file or folder has been restored

Configuring the Email Notification Interval

In addition to enabling and disabling email notifications, bulk email notifications can be configured to be sent out: *As soon as possible* (during the next cron execution), *Hourly*, *Daily*, and *Weekly*. To do so, pick the interval in the "*Send emails:*" drop-down field at the bottom of the Activity configuration panel.

Using the Calendar App

The Calendar app is not enabled by default in ownCloud and needs to be enabled separately. You can download it via [the market app](#).

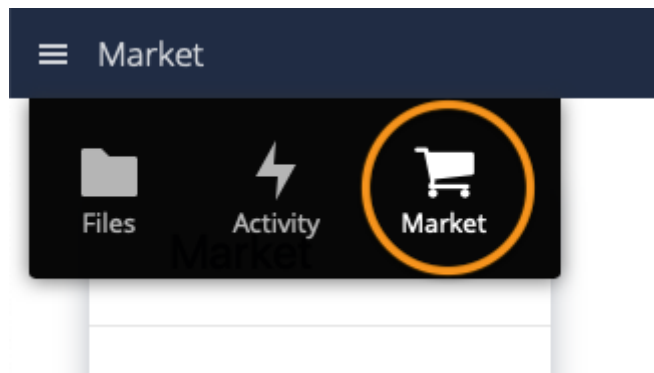
Using the Contacts App

The Contacts app is not enabled by default in ownCloud and needs to be enabled separately. You can download it via [the market app](#).

The Market App

Log in to the Marketplace From the Market App

To log in to [the ownCloud Marketplace](#) from the Market app, open the Market app by clicking on the Market app icon in the top-level ownCloud navigation menu, which you can see above.



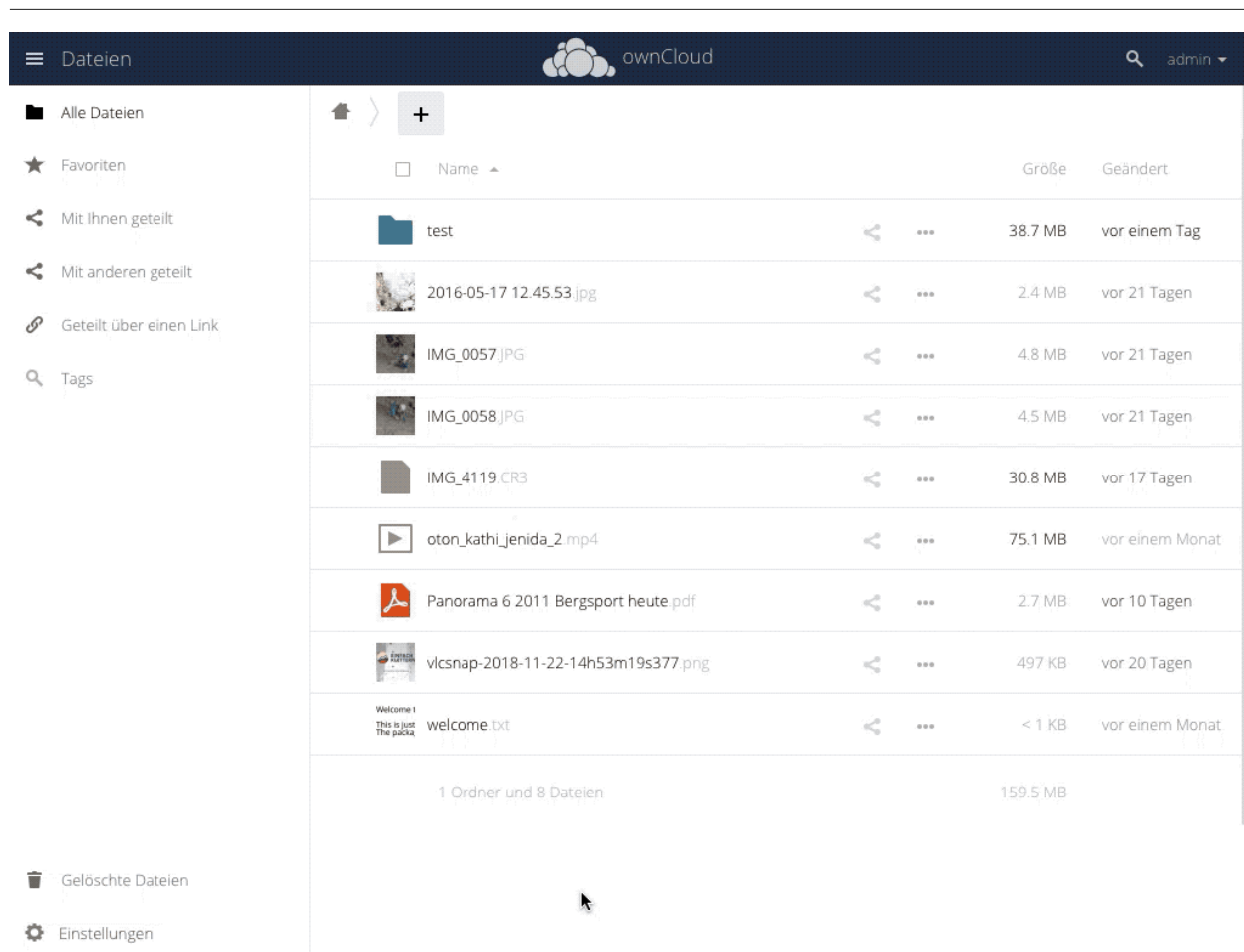
Once on the Market app, click [**LOGIN**], located at the bottom of the left-hand side navigation menu.

You are next asked to grant the Market app access to your Marketplace account. Enter your Marketplace username and password and click [**LOGIN**].

Next, confirm your request to grant the Market app access to your Marketplace account. To give this confirmation, click [**GRANT ACCESS**].

Once you've done this, you are redirected back to the Market app in your ownCloud installation. The original "*LOGIN*" button is now greyed out and labelled "*LOGGED IN*".

The Media Viewer App



Introduction

The **Media Viewer app** is a lightweight viewer for pictures and videos which integrates with the files app, and is released under the GPLv2. It replaces the gallery and files_videoplayer apps, which have now been deprecated, and supports the same basic feature set as the deprecated apps.

It supports the following functionality:

- **Image:** *preview, zoom, rotation, and download*
- **Video:** *playback and download*



The app will support paginating through all media files in the current directory, even if only one media file was chosen to be previewed.

Features

- Support for a large selection of image and video formats (depending on server setup)
- Fullscreen, zoomable slideshow view integrated with the Files view and Public Links
- Image rotation
- Sort images by name or date
- Image and video download straight from the slideshow
- Native SVG support
- Mobile support

Supported File Formats

- The supported video formats depend on the user's browser. However, the app supports *MP4*, *Ogg*, and *WebM*.
- The supported image formats depend on the server capabilities

Use Redis for Files Locking

Using Redis for files locking improves app performance by a factor of 10, when loading an album.

Unsupported File Formats

- Support for playing Apple QuickTime (*.mov) does not work in Chrome - however it is supported in Safari and Mozilla.

The Admin Audit App

Introduction

The Admin Audit app is an auditing module for ownCloud to trace the actions of users and administrators.

Configuration

The following configuration is required, in "config.php", to redirect audit messages into a log file.

```
'log.conditions' => [  
  [  
    'apps' => ['admin_audit'],  
    // Adjust the path below, to match your setup  
    'logfile' => '/var/www/owncloud/data/admin_audit.log'  
  ]  
]
```

Please note that the target path must be writeable for the webserver user. All messages, regardless of log level, will be logged there. To ignore all CLI triggered events (not the default), you can set the following option:

```
{occ-command-example-prefix} config:app:set admin_audit ignore_cli_events  
--value='yes'
```

Grouped Logging

With each log message, several users are calculated to be the 'audit context'. This is the list of users which are related to the log message. Additionally, each log message includes a list of groups that the users are a member of, to enable filtering and splitting of the log messages at a later date. In cases when users are members of many groups, to reduce the data output, the group list can be filtered using the following config option:

```
'admin_audit.groups' => [
  'group1',
  'group2'
]
```

When the filter is configured, only the filtered list of groups will be output in **auditGroups**, else, all groups that the **auditUsers** are a member of are output.

Connect with Splunk Cloud

Splunk captures, indexes, and correlates real-time data in a searchable repository from which it can generate graphs, reports, alerts, dashboards, and visualizations.

— <https://en.wikipedia.org/wiki/Splunk>, Wikipedia

To connect ownCloud with **Splunk Cloud**, you need to carry out the steps below:

1. Install the *Splunk Universal Forwarder*



You can find the system requirements for installing Splunk in [the Splunk documentation](#).

2. Connect your local forwarder to your Splunk Cloud instance You can do this by running the following command:

```
# Change the URL to match your setup
splunk set deploy-poll input-prd-your-server-here.cloud.splunk.com:8089
```

3. Install the *Splunk Cloud* credentials You can do this by running the following command:

```
# Change the path and admin setting to match your setup
splunk install app path/to/splunkclouduf.spl -auth admin:changeme
```

4. Set the ownCloud audit log to be monitored by Splunk You can do this by adding the following configuration to **inputs.conf**:

```
# The file, including the path, must match the 'log.conditions' setting above
[monitor://var/www/owncloud/data/admin_audit.log]
disabled = false
sourcetype = _json
index = main
```

5. Configure **props.conf**, as in the following configuration example, to ensure that the time field is correctly used and that the fields are extracted

```
[_json]
INDEXED_EXTRACTIONS = json
KV_MODE = json
TIMESTAMP_FIELDS = [Time]
category = Structured
```



For more details on configuring the Splunk Universal Forwarder, please refer to [Splunk's online documentation](#).

Output

The audit app listens for internal ownCloud events and hooks and produces a rich set of audit entries useful for reporting on the usage of your ownCloud server. Log entries are based upon the internal ownCloud logging system but contain extra fields that hold relevant data fields related to the specific event.

Each event will contain, at a minimum, the data in the following table:

Setting	Type	Description
<code>remoteAddr</code>	string	The remote client IP
<code>user</code>	string	The UID of the user performing the action. Or "IP x.x.x.x.", "cron", "CLI", "unknown"
<code>url</code>	string	The process request URI
<code>method</code>	string	The HTTP request method
<code>userAgent</code>	string	The HTTP request user agent
<code>time</code>	string	The time of the event e.g.,: 2018-05-08T08:26:00+00:00
<code>app</code>	string	Always 'admin_audit'
<code>message</code>	string	Sentence explaining the action
<code>action</code>	string	Unique action identifier e.g.,: file_delete or public_link_created
<code>CLI</code>	boolean	If the action was performed from the CLI
<code>level</code>	integer	The log level of the entry (usually 1 for audit events)

Please refer to the follow-on sections to see the event- and hook-specific data that is returned.

- [Apps](#)
- [Auth](#)
- [Comments](#)
- [Config](#)
- [Console](#)
- [Custom Groups](#)
- [File Lifecycle](#)
- [Files](#)
- [Holding Period](#)

- Impersonate
- SMB ACL
- Sharing
- Tags
- User Preference
- Users

Apps

	This is an enterprise app
---	---------------------------

app_enabled

Setting	Type	Description
targetApp	string	The app ID of the enabled app.
groups	string[]	Array of group IDs if the app was enabled for certain groups.

app_disabled

Setting	Type	Description
targetApp	string	The app ID of the disabled app.

Auth

user_login

Setting	Type	Description
success	boolean	If the login was successful.
login	string	The attempted login value.

user_logout

Comments

All comment events have the same data:

Setting	Type	Description
commentId	string	The comment identifier.
path	string	The path to the file that the comment is attached to.
fileId	string	The file identifier.

Config

config_set

Setting	Type	Description
settingName	string	The key.

Setting	Type	Description
settingValue	string	The new value.
oldValue	string	The old value.
created	boolean	If the setting is created for the first time.

config_delete

Setting	Type	Description
settingName	string	The key.

Console

command_executed

Setting	Type	Description
command	string	The exact command that was executed.

Custom Groups

custom_group_member_removed

Setting	Type	Description
removedUser	string	The UID of the user that was removed from the group.
group	string	The custom group name.
groupId	integer	The custom group id.

custom_group_user_left

Setting	Type	Description
removedUser	string	The UID of the user that left the group.
group	string	The custom group name.
groupId	integer	The custom group id.

custom_group_user_role_changed

Setting	Type	Description
targetUser	string	The UID of the user that changed role.
group	string	The custom group name.
groupId	integer	The custom group id

Setting	Type	Description
roleNumber	integer	The new role number. <ul style="list-style-type: none"> • 0 = member • 1 = admin

custom_group_renamed

Setting	Type	Description
oldGroup	string	The old custom group name.
group	string	The new custom group name.
groupId	integer	The custom group id

custom_group_created

Setting	Type	Description
group	string	The custom group name created.
groupId		The custom group id.
addedUser	string	The UID of the user added.
admin	boolean	

File Lifecycle

	requires at least version 1.0.0.
---	----------------------------------

lifecycle_archived

Setting	Type	Description
path	string	The path to the file that was archived
owner	string	The UID of the owner of the file that was deleted
fileId	integer	The file ID for the file that was archived

lifecycle_restored

Setting	Type	Description
path	string	The path to the file that was restored
fileId	integer	The number of days interval specified during expiration

lifecycle_expired

Setting	Type	Description
fileId	integer	The file id of the file that was expired

Files

file_create

Setting	Type	Description
path	string	The full path to the create file.
owner	string	The UID of the owner of the file.
fileId	string	The newly created files identifier.

file_read

Setting	Type	Description
path	string	The full path to the file.
owner	string	The UID of the owner of the file.
fileId	string	The files identifier.

file_update

Setting	Type	Description
path	string	The full path to the updated file.
owner	string	The UID of the owner of the file.
fileId	string	The updated files identifier.

file_delete

Setting	Type	Description
path	string	The full path to the updated file.
owner	string	The UID of the owner of the file.
fileId	string	The updated files identifier.

file_copy

Setting	Type	Description
oldPath	string	The full path to the source file.
path	string	The full path to the new file.
sourceOwner	string	The UID of the owner of the source file.
owner	string	The UID of the owner of the file.
sourceFileId	string	The source files identifier.
fileId	string	The new files identifier.

file_rename

Setting	Type	Description
oldPath	string	The original path file.
path	string	The new path file.
fileId	string	The files identifier.

file_trash_delete

Setting	Type	Description
owner	string	The UID of the owner of the file.
path	string	The full path to the deleted file.

file_trash_restore

Setting	Type	Description
owner	string	The UID of the owner of the file.
fileId	string	The restored files identifier.
oldPath	string	The original path to the file.
newPath	string	The new path to the file.
owner	string	The UID of the owner of the file.


file_version_delete

Setting	Type	Description
path	string	The full path to the version file deleted.
trigger	string	The delete trigger reasoning.

file_version_restore

Setting	Type	Description
path	string	The full path to the file being restored to the new version.
revision	string	The revision of the file restored

Holding Period

	requires at least v0.1.3.
---	---------------------------

Impersonate

impersonated

Setting	Type	Description
user	string	The current user who did an impersonate action.
targetUser	string	The user who is being impersonated.

impersonate_logout

Setting	Type	Description
<code>user</code>	string	The user who performed impersonate action.
<code>targetUser</code>	string	The user who was being impersonated.

Sharing

Sharing events come with a default set of fields:

Setting	Type	Description
<code>fileId</code>	string	The file identifier for the item shared.
<code>owner</code>	string	The UID of the owner of the shared item.
<code>path</code>	string	The path to the shared item.
<code>shareId</code>	string	The sharing identifier. It is not available for <code>public_link_accessed</code> or when recipient unshares.

file_shared

Setting	Type	Description
<code>itemType</code>	string	<code>file</code> or <code>folder</code>
<code>expirationDate</code>	string	The text expiration date in format: <code>yyyy-mm-dd</code>
<code>sharePass</code>	boolean	If the share is password protected.
<code>permissions</code>	string	The permissions string e.g.,: "READ"
<code>shareType</code>	string	<code>group</code> <code>user</code> or <code>link</code>
<code>shareWith</code>	string	The UID or GID of the share recipient. (not available for public link)
<code>shareOwner</code>	string	The UID of the share owner.
<code>shareToken</code>	string	For link shares the unique token, else null

file_unshared

Setting	Type	Description
<code>itemType</code>	string	<code>file</code> or <code>folder</code>
<code>shareType</code>	string	<code>group</code> <code>user</code> or <code>link</code>
<code>shareWith</code>	string	The UID or GID of the share recipient.

share_permission_update

Setting	Type	Description
<code>itemType</code>	string	<code>file</code> or <code>folder</code>
<code>shareType</code>	string	<code>group</code> <code>user</code> or <code>link</code>
<code>shareOwner</code>	string	The UID of the share owner.

Setting	Type	Description
permissions	string	The new permissions string e.g.,: "READ"
shareWith	string	The UID or GID of the share recipient. (not available for public link)
oldPermissions	string	The old permissions string e.g.,: "READ"

share_name_updated

Setting	Type	Description
oldShareName	string	The previous share name.
shareName	string	The updated share name.

share_password_updated

Setting	Type	Description
itemType	string	file or folder
shareOwner	string	The UID of the share owner.
permissions	string	The full permissions string e.g.,: "READ"
shareToken	string	The share token.
sharePass	boolean	If the share is password protected.

share_expiration_date_updated

Setting	Type	Description
itemType	string	file or folder
shareType	string	group user or link
shareOwner	string	The UID of the owner of the share.
permissions	string	The permissions string e.g.,: "READ"
expirationDate	string	The new text expiration date in format: yyyy-mm-dd
oldExpirationDate	string	The old text expiration date in format: yyyy-mm-dd

share_accepted

Setting	Type	Description
itemType	string	file or folder.
path	string	The path of the shared item.
owner	string	The UID of the owner of the shared item.
fileId	string	The file identifier for the item shared.

Setting	Type	Description
shareId	string	The sharing identifier. This is not available for public_link_accessed.
shareType	string	group user

share_declined

Setting	Type	Description
itemType	string	file or folder.
path	string	The path of the shared item.
owner	string	The UID of the owner of the shared item.
fileId	string	The file identifier for the item shared.
shareId	string	The sharing identifier. This is not available for public_link_accessed.
shareType	string	group user

federated_share_received

Setting	Type	Description
name	string	The path of shared item
targetuser	string	The target user who sent the item
shareType	remote	string

federated_share_accepted

Setting	Type	Description
itemType	string	The path of shared item
targetUser	string	The target user who sent the item
shareType	string	remote

federated_share_declined

Setting	Type	Description
itemType	string	The path of shared item
targetuser	string	The target user who sent the item
shareType	string	remote

public_link_accessed

Setting	Type	Description
shareToken	string	The share token.
success	boolean	If the request was successful.
itemType	string	file or folder

public_link_removed

Setting	Type	Description
shareType	string	link

public_link_accessed_webdav

Setting	Type	Description
token	string	The token used to access the url.

federated_share_unshared

Setting	Type	Description
targetUser	string	The user who initiated the unshare action
targetmount	string	the file/folder unshared.
shareType	string	remote

SMB ACL

before_set_acl

Setting	Type	Description
user	string	The user who is trying to set the ACL
ocPath	string	The owncloud instance path
smbPath	string	The SMB path
descriptor	array	<p>The descriptor array. It contains to following keys:</p> <ul style="list-style-type: none">• revision - integer - Always 1• owner - string - The SMB owner• group - string - The SMB group• acl - array - A list of ACEs. The list could be empty. Each ACE contains<ul style="list-style-type: none">◦ trustee - string - The SMB user affected by this ACE◦ mode - string - "allowed" or "denied"◦ flags - string - Inheritance flags◦ mask - string - Permission mask◦ flagsAsInt - integer - The inheritance flags as integer value◦ maskAsInt - integer - The permission mask as integer value

after_set_acl

Setting	Type	Description
user	string	The user who is trying to set the ACL

Setting	Type	Description
ocPath	string	The owncloud instance path
smbPath	string	The SMB path
descriptor	array	<p>The descriptor array. It contains to following keys:</p> <ul style="list-style-type: none"> • revision - integer - Always 1 • owner - string - The SMB owner • group - string - The SMB group • acl - array - A list of ACEs. The list could be empty. Each ACE contains <ul style="list-style-type: none"> ◦ trustee - string - The SMB user affected by this ACE ◦ mode - string - "allowed" or "denied" ◦ flags - string - Inheritance flags ◦ mask - string - Permission mask ◦ flagsAsInt - integer - The inheritance flags as integer value ◦ maskAsInt - integer - The permission mask as integer value

Tags

tag_created

Setting	Type	Description
tagName	string	The tag name.

tag_deleted

Setting	Type	Description
tagName	string	The tag name.

tag_updated

Setting	Type	Description
oldTag	string	The old tag name.
tagName	string	The new tag name.

tag_assigned

Setting	Type	Description
tagName	string	The tag name.
fileId	string	The file identifier to which the tag was assigned.
path	string	The path to the file.

tag_unassigned

Setting	Type	Description
tagName	string	The tag name.
fileId	string	The file identifier from which the tag was unassigned.
path	string	The path to the file.

User Preference

update_user_preference_value

Setting	Type	Description
key	string	The key
value	string	The value associated with the key
appname	string	The name of the app
user	string	The UID of the user who has the preference key-value for the app.

user_preference_set

Setting	Type	Description
key	string	The key
value	string	The value associated with the key
appname	string	The name of the app
user	string	The UID of the user who has the preference key-value for the app.

remove_user_preference_key

Setting	Type	Description
key	string	The key
appname	string	The name of the app
user	string	The UID of the user whose preference key is deleted for the app.

remove_preferences_of_user

Setting	Type	Description
user	string	The UID of the user, whose all user preferences are deleted.

delete_all_user_preference_of_app

Setting	Type	Description
appname	string	The name of the app whose all user preferences are deleted.

Users

user_created

Setting	Type	Description
targetUser	string	The UID of the created user.

user_password_reset

Setting	Type	Description
targetUser	string	The UID of the user.

group_member_added

Setting	Type	Description
targetUser	string	The UID of the user.
group	string	The GID of the group.

user_deleted

Setting	Type	Description
targetUser	string	The UID of the user.

group_member_removed

targetUser	string	The UID of the user.
group	string	The GID of the group.

user_state_changed

Setting	Type	Description
targetUser	string	The UID of the user.
enabled	boolean	If the user is enabled or not.

group_created

Setting	Type	Description
group	string	The GID of the group.

group_deleted

Setting	Type	Description
group	string	The GID of the group.

user_feature_changed

Setting	Type	Description
targetUser	string	The UID of the user.

Setting	Type	Description
group	string	The GID of the group (or empty string).
feature	string	The feature that was changed.
value	string	The new value.

Contacts & Calendar

The Contacts, Calendar, and Mail apps are not included in ownCloud 9 and later, and are not supported. You can install them from the ownCloud Marketplace by clicking **[Install]** on their respective entries. You can find them under **Market > Productivity**.

iOS - Synchronize iPhone/iPad

Calendar

1. Open the settings application.
2. Select **Mail > Contacts > Calendars**.
3. Select **[Add Account]**.
4. Select **[Other]** as account type.
5. Select **[Add CalDAV]** account.
6. For server, type example.com/remote.php/dav/principals/users/USERNAME/
7. Enter your user name and password.
8. Select Next.
9. If your server does not support SSL, a warning will be displayed. Select **[Continue]**.
10. If the iPhone is unable to verify the account information perform the following steps:
 - Select **[OK]**.
 - Select **[Advanced Settings]**.
 - If your server does not support SSL, make sure Use SSL is set to OFF.
 - Change port to 80.
 - Go back to account information and click **[Save]**.

Your calendar will now be visible in the Calendar application

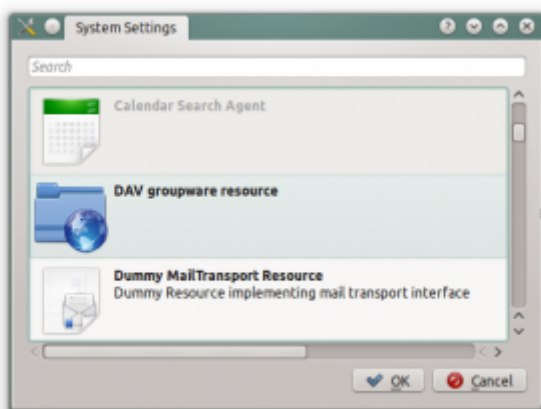
Address book

1. Open the settings application.
2. Select **Mail > Contacts > Calendars**.
3. Select **[Add Account]**.
4. Select Other as account type.
5. Select **[Add CardDAV]** account.
6. For server, type example.com/remote.php/dav/principals/users/USERNAME/
7. Enter your user name and password.
8. Select Next.

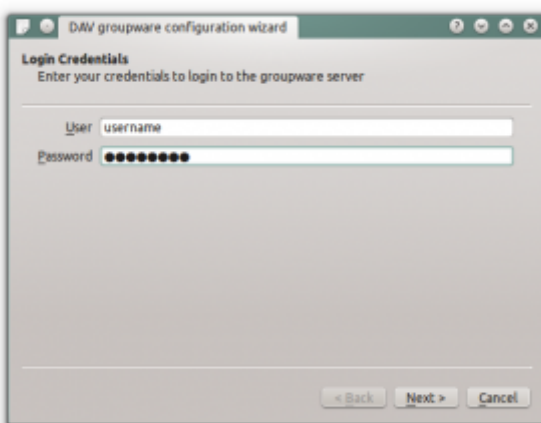
-
9. If your server does not support SSL, a warning will be displayed. Select **[Continue]**.
 10. If the iPhone is unable to verify the account information perform the following:
 - Select **[OK]**.
 - Select advanced settings.
 - If your server does not support SSL, make sure Use SSL is set to OFF.
 - Change port to 80.
 - Go back to account information and click **[Save]**.

Now should now find your contacts in the address book of your iPhone. If it's still not working, have a look at the [Troubleshooting Contacts & Calendar](#) guides.

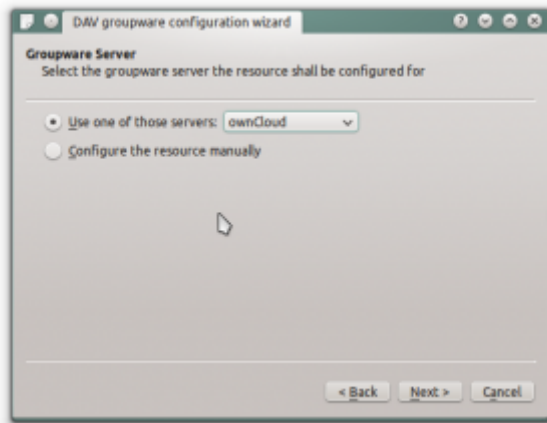
Synchronizing with KDE SC



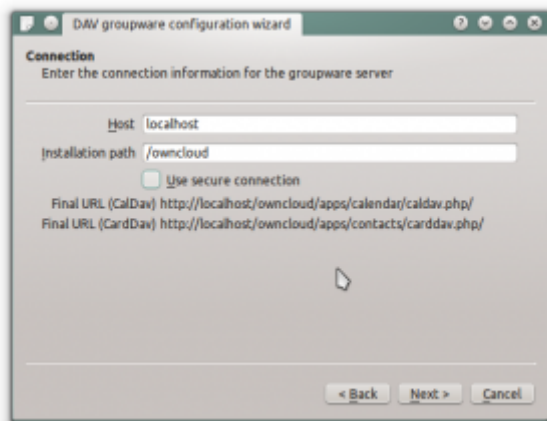
From KDE SC 4.8 and forward setting up ownCloud is very easy. Note that the KDE calendar needs to have the ownCloud Calendar and Contacts apps enabled on the ownCloud server. You need both and not just the Calendar. From System Settings Personal Information/Akonadi Resources Configuration select DAV Groupware resource.



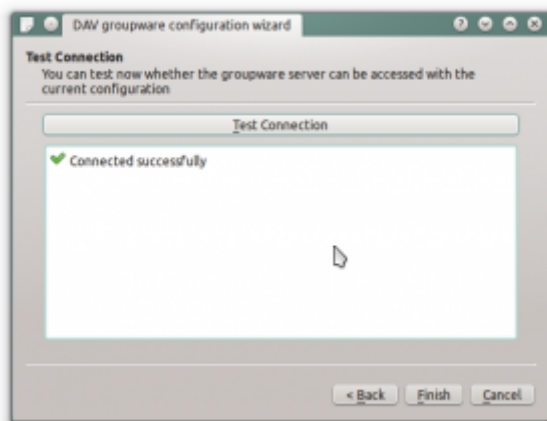
Enter your ownCloud username and password and click **[Next]**.



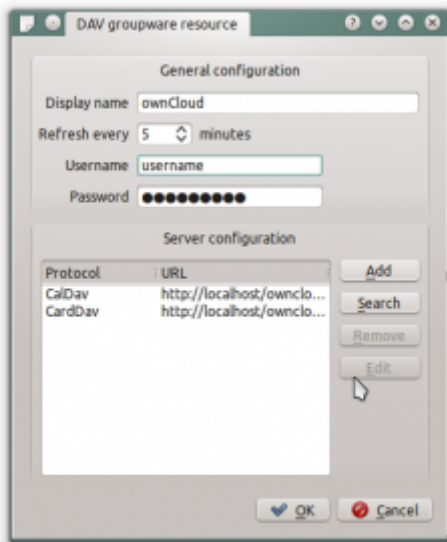
Select ownCloud in the drop down list and click [Next].



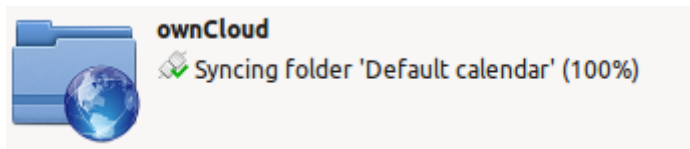
Enter the host name and installation path. If you do not use SSL remember to de-select "Use secure connection".



Test the connection. If everything went well you should see a message like the one below.

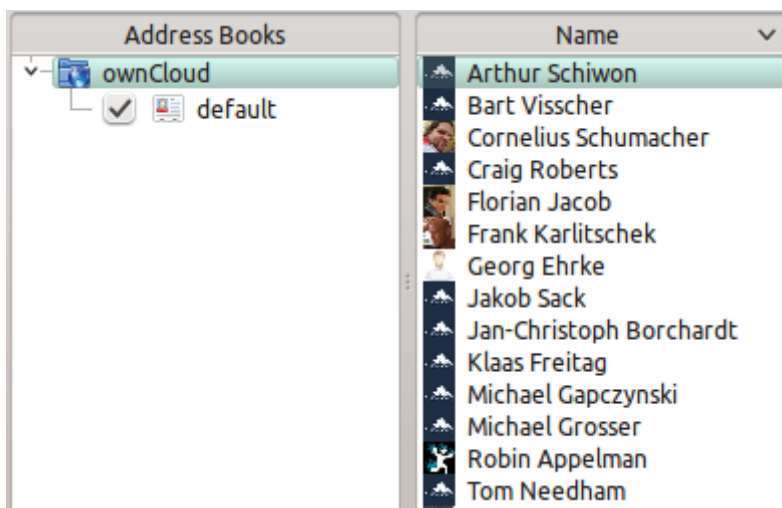
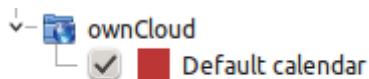


Click "Finish" and you will be able to change the display name and refresh interval.



Now you should see the Akonadi resource doing the first synchronization.

You can find the Contacts and Calendars in Kontact (or KOrganizer/KAddressbook if you run the programs separately.)



Synchronizing with OS X

To use ownCloud with iCal you will need to use the following URL:

```
https://example.com/remote.php/dav/principals/users/USERNAME/
```

The setup is basically the same as with iOS using the path

<https://example.com/remote.php/dav/principals/users/USERNAME/> to sync with ownCloud. For OS X 10.7 Lion and 10.8 Mountain Lion everything works fine, but OS X 10.6 (Snow Leopard) and older needs some fiddling to work. A user contributed the following:

1. Make sure, addressbook is not running. If it is, select the windows and press *Command+Q* to terminate it.
2. Navigate to **Users > YOUR_USERNAME > Library > Application Support > AddressBook > Sources**. If you already have some kind of addressbook setup, it is likely you will see some folders named like this **BEA92826-FBF3-4E53-B5C6-ED7C2B454430**. Note down what folders there are now and leave the window open.
3. Open addressbook and try to add a new CardDav addressbook. At this point, it does not matter what information you enter. It will come up with the same error message you mentioned before when you click **[Create]**. Ignore it and click **[Create]** again. A non-functional addressbook will be added.
4. Close addressbook again using *Command+Q*
5. Go back to the folder window from step 2. You will now see a newly created folder with another long string as its name.
6. Navigate to the newly created folder and edit the **Configuration.plist** with your favorite text editor.
7. Search for a section looking like this:

```
<key>servername</key> <string>https://0(null)</string>
<key>username</key> <string>Whatever_you_entered_before</string>
```

8. Make it look like this. Please note that the :443 after **example.com** is important:

```
<key>servername</key>
<string>https://example.com:443/owncloud/remote.php/dav/principals/users/US
ERNAME</string> <key>username</key> <string>username</string>
```

9. Save the file and open addressbook again. It will not work yet.
10. Open the preferences for your ownCloud CardDAV-Account and enter your password.
11. You may have to restart addressbook once more. After this, it should work.

If it's still not working, have a look at the [Troubleshooting Contacts & Calendar](#) guides.

There is also an easy [HOWTO](#) in the forum.

Thunderbird - Synchronize Addressbook

As someone who is new to ownCloud, New to SoGo Connector, and new to Thunderbird Addressbook, here is what you need in excruciating pithy detail to make this work (for all the other lost souls out there):

1. [Thunderbird](#) for your OS unless it comes with your OS distribution (Linux)
2. [Sogo Connector](#) (latest release)
3. [Lightning](#) (a Thunderbird calendar add-on. At the time of writing, syncing your

contacts only works with this add-on installed.)

With an installed Thunderbird mail tool, an installed SoGo Connector, and an installed Lightning add-on:

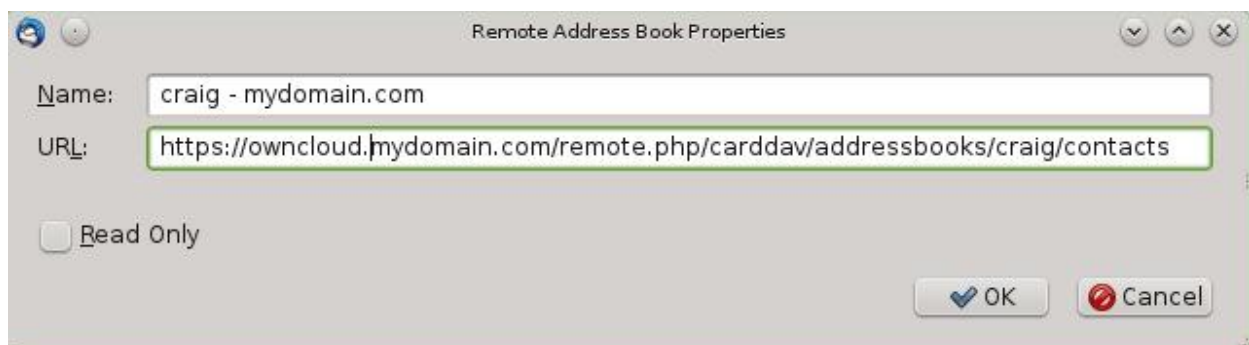
1. Thunderbird Addressbook is in the Thunderbird "Tools" Menu
2. In the Thunderbird Addressbook application:
 - **File > New > Remote Addressbook** (SoGo Connector added this)
 - "**Name:**" is the name you want to give your Addressbook in the Thunderbird addressbook bar area
 - "**URL:**" is found in your ownCloud Contacts area, click the little [gear symbol]



in the -bottom left- of the Contacts View (same symbol as found in the -top right- in the Calendar view). Then look for a little impeller symbol



which will display the URL you need for your installation to work.



Once installed, synchronize (right click on your newly made remote address book and select [**synchronize**]). You'll see your address book populate from ownCloud! **Don't** click [**read only**] above unless you don't want to modify your ownCloud server addressbook, like it contains a listing of corporate contacts and is shared with lots of people, and you don't want a new user dragging it somewhere unintended.

The rest of the details of dealing with Thunderbird addressbook are left to the reader... First thing I learned is dragging a contact to a different addressbook is a move operation. If you are worried about losing the contact, save it to a VCF file using ownCloud (Or LDIF using Thunderbird Addressbook) first! Like dragging from the ownCloud Addressbook to the Personal Address Book removes the contact from ownCloud Server (*deleting it from all the other synchronized installations*) and puts it in your Local Machine -only-Address Book. So be careful or you'll have unintended consequences where you might have intended a copy operation.

Contact Pictures are also sync'ed!

Troubleshooting

General Troubleshooting

This section contains a list of error messages that ownCloud displays when things go wrong. For each message, you can find the most likely cause, a resolution, and one or more translations.

Problem loading page, reloading in 5 seconds

Cause	
Resolution	
Translation	Problem beim Laden der Seite, Seite wird in 5 Sekunden nochmals geladen

Already existing files

Cause	
Resolution	
Translation	Bereits existierende Dateien

Which files do you want to keep?

Cause	
Resolution	
Translation	Welche Dateien möchten Sie behalten?

Error while sharing

Cause	
Resolution	
Translation	Fehler beim Teilen

Error while unsharing

Cause	
Resolution	
Translation	Fehler beim Aufheben der Freigabe

Error while sending notification

Cause	
Resolution	
Translation	Fehler beim Senden der Benachrichtigung

Resharing is not allowed

Cause	
-------	--

Resolution	
Translation	Das Weiterverteilen ist nicht erlaubt

Error removing share

Cause	
Resolution	
Translation	Fehler beim Entfernen der Freigabe

Internal Server Error

Cause	
Resolution	
Translation	Interner Serverfehler

This ownCloud instance is currently in single user mode.

Cause	
Resolution	
Translation	Diese ownCloud-Instanz befindet sich derzeit im Einzelbenutzermodus.

This %s instance is currently in maintenance mode, which may take a while.

Cause	
Resolution	
Translation	Diese %s-Instanz befindet sich gerade im Wartungsmodus, was eine Weile dauern kann.

File Troubleshooting

Listed here are the most common errors you may encounter while attempting to upload files, along with what they mean, and possible workarounds.

Error while copying file to target location (copied bytes: xxx, expected filesize: yyy)

This error is most likely due to an issue with the target storage location. During file uploads the file data is read from PHP input and copied into a part file on the target storage.

If the target storage is not local (eg: FTP) and that storage is slow, not available, or broken it is likely that the operation will fail either at the beginning, or in the middle of the copy. Other reasons for this message can be that, when writing to external storage, the connection took too long to respond or the network connection was flaky.

Sharing sidebar does not show **Shared with you by ...** for remote shares

In some scenarios, when users share folders and files with each other they cannot be scanned. There are a variety of reasons why this happens, which can include firewalls and broken servers.

In these situations, when the initial scan did not complete successfully, the mount point cannot appear in the ownCloud web UI. This is because ownCloud was not able to generate a matching file cache entry, nor retrieve any metadata about whether it's a folder or file (mime type), etc.

PIM Troubleshooting

BlackBerry OS up to 10.2.2102 does not accept a URL with protocol **https://** in front of the server address. It will always tell you that it cannot login on your server. So instead of writing:

```
https://example.com/remote.php/dav/principals/users/USERNAME/
```

in the server address field, you have to write:

```
example.com/remote.php/dav/principals/users/USERNAME/
```

Have You Found a Mistake In The Documentation?

If you have found a mistake in the documentation, no matter how large or small, please let us know by [creating a new issue in the docs repository](#).