

Plugin *OwnCloud*

Documentation

Date of Release	11.04.2016
Document-Version	2.0
Plugin-Version:	2.0.0
Author:	Theodor Truffer

Änderungsprotokoll

Version 1.0, 11.04.2016

ttx

Version 2.0, 30.05.2017

ttx

- created
- OAuth2 supported

Table of contents

1 Introduction.....	4
1.1 Supported ILIAS-Versions.....	4
2 Feature Overview.....	4
3 Installation.....	4
3.1 OwnCloud - Configuration.....	4
3.2 OAuth2-Authenticaiton.....	5
3.3 ILIAS – Configuration.....	5
4 Usage of the Plugin.....	5
4.1 Create Object.....	5
4.2 Configure Object.....	5
4.3 Browse OwnCloud.....	6

1 Introduction

OwnCloud is a CloudHook plugin which offers the possibility to bind the Cloud Service OwnCloud to ILIAS. It allows users to manage the files and folder located in the Cloud directly within ILIAS.

1.1 Supported ILIAS-Versions

The plugin supports all ILIAS-Versions from 5.0.x to 5.2.x.

2 Feature Overview

PL-Version	Changes
1.0.0	Key-Features: <ul style="list-style-type: none"> Creating a Cloud Object of type <i>OwnCloud</i> Perform Base-Commands inside the OwnCloud folder (upload, download, create, delete) Change root folder Choose root folder at creation <p>Support for ILIAS 5.1</p>
2.0.0	Key-Features: <ul style="list-style-type: none"> Authentication via OAuth2 protocol Support for ILIAS 5.0 - 5.2

3 Installation

To install this plugin, change to your ILIAS-Root-Directory and execute the following commands:

```
mkdir -p Customizing/global/plugins/Modules/Cloud/CloudHook/
cd Customizing/global/plugins/Modules/Cloud/CloudHook/
git clone https://github.com/studer-raimann/OwnCloud.git
```

As ILIAS administrator go to "Administration->Plugins" and install/activate the plugin.

3.1 OwnCloud - Configuration

So far the plugin is not working. In order to work, the plugin needs to be properly configured. To do so, navigate to „Administration->Plugins“, find the entry for the OwnCloud plugin and choose *Configure* from the *Actions*-Menu. There you can choose a title and info text for the OwnCloud service (which should already be set), and the URL of your OwnCloud server (something like <https://cloud.mydomain.com>). Below you can enter an alternativ WebDAV path. The default is

'remote.php/webdav' which fits in most cases and don't have to be changed. So if you're not sure, try to enter nothing here.

If you wish to authenticate the plugin via the OAuth2 protocol, please read the next chapter (3.2). If you choose to authenticate via Basic Authentication then this chapter can be skipped.

3.2 OAuth2-Authenticaiton

If you wish to authenticate the plugin via the OAuth2 protocol, activate the checkbox '*OAuth2-Authentication*' in the plugin configuration and fill in the *Client Identifier* and *Secret*. To receive these informations, you have to install the oauth2 app on your OwnCloud server and register your ILIAS installation in the app. To do so, visit the app's repository at <https://github.com/owncloud/oauth2> and follow the chapters '*Installing the app*' and '*Protocol Flow → 1. Client registration*'. Note that while registering the client, the *redirection URL* must look like this (replace <https://ilias.mydomain.ch> with your ILIAS domain):

<https://ilias.mydomain.ch/Customizing/global/plugins/Modules/Cloud/CloudHook/OwnCloud/redirect.php>

3.3 ILIAS – Configuration

To be able to create an OwnCloud Object, an administrator has to enable the creation of Cloud-Objects in general. To do so, navigate in the main menu to Administration > Repository. There change to the tab Repository Object Types, find the entry for Cloud Object and set the checkbox that says Enable Creation, on the right side of the entry.

4 Usage of the Plugin

4.1 Create Object

To create a Cloud Object of type OwnCloud, just navigate to the place in the repository where you would like to add the object, click on Add New Item and choose *Cloud Object*. Inside the appearing mask, enter a title and choose *ownCloud* (or the title you defined in the plugin configuraiton) as service.

4.2 Configure Object

Once created, the object is not able to connect to the OwnCloud service yet. If you're using OAuth2-Authentication, you will be redirected to OwnCloud to enter your username and password. If you're using Basic-Authentication (OAuth2 deactivated), you'll have to configure the username and password of a registered OwnCloud user inside the settings of each cloud object.

Note that the password will be stored in *plain text* in the database. If you don't wish to do that, switch to OAuth2-Authentication.

Further you can change the root folder of your OwnCloud object. Just type a path of folders from your OwnCloud and the users will see only subfolders of this path. This setting is very useful, since you can show only a specific subtree of your whole OwnCloud directory.

4.3 Browse OwnCloud

Inside the created object, change to the tab Content to browse your OwnCloud folder. The usage is straightforward: Add new files and folder via the Add New Item button, delete them through the Action menu, open folders and download files by simply clicking on their names.