

# ownCloud 6 auf Ubuntu / Plesk

Installieren, Konfigurieren und Nutzen – am Server und am Client.

## Inhaltsverzeichnis

1. OwnCloud (Server) holen und installieren.....	2
1.1 Vorbereitung der Domain-Hosting.....	2
1.2 Installation von ownCloud auf dem Server.....	2
1.3 Gegebenenfalls WebDAV in Apache ausschalten.....	2
1.4 Gegebenenfalls Nginx-Konfiguration einpflegen.....	3
1.5 Erstkonfiguration der ownCloud-Installation.....	3
1.6 Apps aktivieren bzw. installieren.....	3
2. Dateien am Server speichern und verwalten.....	4
2.1 Grundsätzliches zur Arbeitsweise von ownCloud.....	4
2.2 Die richtige Strategie entwickeln.....	4
2.2.1 Dateien: Browser / WebDAV / Synchronisierung Lokal <=> Server.....	5
2.2.2 Benutzer und Gruppen in ownCloud.....	6
2.2.3 Andere Datenquellen in ownCloud einhängen („Mounten“)......	7
2.2.3.1 Lokale Plattenbereiche - local external storage.....	7
2.2.3.2 Wie ownCloud Änderungen an Fremdbeständen (nicht) merkt.....	8
2.2.4 Der Client – gut, mit kleinen Einschränkungen.....	9
2.3 Benutzer und Gruppen anlegen.....	10
2.4 Dateien hochladen: Die Reihenfolge ist wichtig.....	10
2.5 Den Client konfigurieren.....	10
2.6 Weiterführende Themen.....	11
2.6.1 NetDrive zur Kontrolle konfigurieren.....	11
2.6.2 Andere Arten externer Speicher einhängen.....	12
2.6.3 Tipp: Geteilte Dateien gleichzeitig bearbeiten.....	12
2.6.4 Ignorierte Dateien / Sonderfall .htaccess.....	12
3. Eigene Kalenderdaten am Server halten mittels CalDAV.....	13
4. Eigene Kontaktdaten am Server halten mittels CardDAV.....	13

Holen Sie auch die offizielle [ownCloud-Dokumentation](#).

OwnCloud bietet folgende Dienste:

1. Eigene Dateien am Server speichern und verwalten
2. Eigene Kalenderdaten mittels CalDAV am Server halten
3. Eigene Kontakte mittels CardDAV am Server halten

Dieses Beispiel wurde mit Ubuntu-Server 12.04 LTS und Plesk 11.5 erarbeitet – in anderen Umgebungen müssen Sie ggf. manche Punkte entsprechend anpassen.

# 1. OwnCloud (Server) holen und installieren

ownCloud kann direkt in einer eigenen, dafür reservierten Domain oder Subdomain eingerichtet werden; oder in einem Unterverzeichnis einer vorhandenen Domain. Im Folgenden wird Ersteres beschrieben (Subdomain), da dies eigene Nginx-Einstellungen ermöglicht, so dass die speziellen Nginx-Direktive für ownCloud nicht mit Erfordernissen der Hauptdomain-Konfiguration kollidieren. Außerdem ist so eine bessere Abschottung der Anwendungen untereinander gewährleistet.

## 1.1 Vorbereitung der Domain-Hosting

Anscheinend doch nicht nötig: `cgi.fix_pathinfo = 0`

## 1.2 Installation von ownCloud auf dem Server

Als erstes richten Sie eine neue, leere MySQL-Datenbank für ownCloud ein und notieren Sie die Zugangsdaten.

Zur Installation der ownCloud-Dateien gibt es zwei Alternativen. Gehen Sie zunächst auf [diese Seite](#), und dort:

1. Falls Sie einen eigenen Server betreiben:

Klicken Sie links auf „Tar or Zip file“, laden Sie die gewünschte Fassung herunter, und entpacken Sie sie auf dem Server – falls Ihr ownCloud auf einer eigene (Sub-) Domain installiert wird, dann ins Wurzelverzeichnis der Domain, sonst in ein Unterverzeichnis der Parent-Domain (z.B. „owncloud“).

```
tar -xjf owncloud-6.0.3.tar.bz2 ; mv owncloud/.htaccess owncloud/* .
rmdir owncloud ; rm owncloud-6.0.3.tar.bz2
chown -R <user>:<group> .htaccess *
```

2. Falls ownCloud in einem Webhosting-Paket installiert wird:

Klicken Sie in der Mitte auf „Web Installer“ – Install ownCloud using the Web Installer – und folgen Sie der Kurzanleitung dort.

## 1.3 Gegebenenfalls WebDAV in Apache ausschalten

Plesk bietet neuerdings selbst ein „File sharing“ an, das letztendlich eine WebDAV-Implementierung ist. Hierfür verwendet Plesk das DAV-Modul des Apache Webservers. Die Funktionalität kann sich allerdings nicht mit ownCloud messen.

Folgende Schritte sind nur nötig falls Sie Apache als Webserver für die Domain benutzen; empfohlen wird falls möglich Nginx, der ist schneller!

Das DAV-Modul ist in einem von Plesk verwalteten Apache generell eingeschaltet, würde sich aber mit ownCloud beissen, da ownCloud eine eigene WebDAV-Implementierung mitbringt (das Open-Source Produkt SabreDAV). Um das Plesk-Feature nicht zu kompromittieren muss das DAV-Modul zielgerichtet nur für das ownCloud-Verzeichnis ausgeschaltet werden.

Plesk – Domain – Web Server Settings – Additional Apache directives

*Additional directives for HTTP:*

```
<Directory /var/www/vhosts/{PathToOwnCloudVerzeichnis}>  
  Dav Off  
</Directory>
```

*Additional directives for HTTPS:*

```
<Directory /var/www/vhosts/{PathToOwnCloudVerzeichnis}>  
  Dav Off  
</Directory>
```

## **1.4 Gegebenenfalls Nginx-Konfiguration einpflegen**

Einzelheiten stehen in meinem Dokument:

Server-Tuning-2\_Ubuntu-12.04-LTS\_Plesk-11.5\_Nginx-PHP-FPM

Am Besten als Datei „owncloud.conf“ ins Wurzelverzeichnis der Domain und diese in Plesk includieren.

## **1.5 Erstkonfiguration der ownCloud-Installation**

Einfach ownCloud aufrufen: <https://<Ihr{Sub}Domain>/>

Es wird das Anlegen eines Admin-Kontos angeboten – setzen Sie einen Kontonamen und Passwort Ihrer Wahl hier ein. Tipp: Um die Sicherheit zu erhöhen wählen Sie keinen einfach zu-errathenden Kontonamen wie „admin“.

"Fortgeschritten" anklicken und das Datenverzeichnis prüfen:

`/var/www/vhosts/<Ihr{Sub}Domain>/data` (so lassen)

"Datenbank einrichten" anklicken, "MySQL" wählen, die Daten der DB eingeben, die Sie oben vorab eingerichtet haben. Host = localhost.

"Installation abschließen" - das war's schon.

ownCloud richtet ein Datenverzeichnis für den Benutzer ein, z.B.

`/var/www/vhosts/<Ihr{Sub}Domain>/data/<IhrAdminKontoNamen>`

Ganz wichtig um die Sicherheit der zwischen Clients und Server übertragenen Daten zu gewährleisten (erzwingen):

{Als Admin einloggen} | Administrator (aus Dropdown-Menü oben rechts) | Sicherheit | HTTPS erzwingen (Häkchen setzen)

## **1.6 Apps aktivieren bzw. installieren**

Wie heutzutage zunehmend üblich, lässt sich die Funktionalität von ownCloud durch Plugins erweitern, die hier „Apps“ lauten. ownCloud wird mit einer Anzahl vorinstallierten „internen“ Apps ausgeliefert – also die gebräuchlichsten – wovon die wichtigsten schon vorsorglich aktiviert sind.

Als Administrator einloggen | Unten links auf „+ Apps“ klicken.

Alle in Ihrem ownCloud installierten Apps werden hier aufgelistet, die aktivierten zuerst mit Namen in Fettschrift.

Online gibt es [hier](#) eine Auswahl weiterer Apps von anderen Entwicklern. Ich habe z.B. das App „Files move“ von dort installiert, welches erlaubt, Dateien innerhalb von ownCloud (im Browser) innerhalb des Dateibaums zu verschieben:

- Das gewünschte App herunterladen
- Zum Server nach „owncloud/apps“ hochladen
- Die „+ Apps“ Seite aktualisieren und das neue App dort aktivieren.

P.S. Das App "Files Tree" sieht verlockend aus, bringt's aber momentan nicht so gut – es nimmt eine eigene Spalte für die Anzeige, die Anzeige ist etwas ungenau und zu wenig Responsive.

## 2. Dateien am Server speichern und verwalten

### 2.1 Grundsätzliches zur Arbeitsweise von ownCloud

Im Gegensatz zu herkömmlichen Verfahren wie z.B. FTP oder SCP „sehen“ Sie mit ownCloud nicht direkt auf einen Bereich der Server-Festplatte. Vielmehr führt ownCloud Buch über Ihre Datenbestände in Datenbanktabellen, aus denen er an allen Schnittstellen ein von ihm erzeugtes „**Bild**“ der Dateien präsentiert. Meist sind zwar ähnliche Verzeichnisbäume letztendlich dahinter – irgendwo – aber das Bild, das präsentiert wird, entsteht aus den Daten in der Datenbank und nicht direkt von den Dateien, die „irgendwo“ liegen.

Dementsprechend kommt der Aktualisierung der Daten in den DB-Tabellen eine zentrale Bedeutung zu.

ownCloud bietet nicht nur Zugang zu den Dateien am Server per Browser und/oder WebDAV, sondern auch mittels eines Clients die Möglichkeit, die Dateien zusätzlich lokal (auf dem eigenen PC) vorzuhalten und die Datenbestände in beiden Richtungen zu synchronisieren. Das ermöglicht ein flottes lokales Arbeiten und auch die Möglichkeit ohne Internet-Anschluss weiter zu arbeiten (z.B. im Zug).

Was ownCloud verständlicherweise etwas in die Knie zwingt, ist riesige Mengen kleiner Dateien. Diese kommen vor allem dann vor, wenn man lokal Archive auspackt – sei es Programmpakete wie ownCloud und eGroupware, oder Info-Archive wie Self-HTML. Daher Archive unbedingt auf einem Bereich der lokalen Festplatte auspacken, der **nicht** synchronisiert wird!

### 2.2 Die richtige Strategie entwickeln

Um ownCloud (mit mehr als nur einem Benutzer) nutzen zu können müssen Sie die Benutzerkonten, Gruppen und Verzeichnisse (Mountpoints) anlegen, die Sie für Ihre Zwecke brauchen. Um das zielführend machen zu können, müssen Sie jedoch eine Menge über die Möglichkeiten, Stärken und Schwächen von ownCloud wissen. Im Fol-

genden versuche ich das Entscheidende – was nicht so ohne Weiteres im Handbuch steht – zu vermitteln. Vorweg:

- **Nichts überstürzen** – gehen Sie ruhig, bedacht und Schritt für Schritt vor.
  - Entwickeln Sie unbedingt eine **Gesamtstrategie**, eher Sie überhaupt handeln.
- 1 Ordnen und säubern Sie die Datenbestände, die auf den Server kommen sollen. Entfernen Sie alle entpackten Archive aus den Synchronisierungsbereichen.
  - 2 Zum Säubern gehört das Löschen von Temp-Dateien (wie z.B. „\*.bak“) sowie das Eintragen solcher Namensmuster (zu gegebener Zeit) in der Ignorierliste des own-Cloud-Clients.
  - 3 Erst jetzt, wo Sie den Überblick über die Datenmenge haben, die auf den Server kommen soll, können Sie eine Strategie für Benutzer, Gruppen und ggf. External Storage entwickeln.
  - 4 Nach der Installation von ownCloud, gehen Sie beim Hochladen der Datenbestände „Eins nach dem Anderen“ vor. Paralleles Arbeit am gleichen Datenbestand von mehr als einem PC aus führt in meiner Erfahrung fast immer zu Problemen!
  - 5 Plattenbereiche, die zwischen PC ('s) und Server synchronisiert werden sollen, zunächst vom „besten“ Ursprungsort hochladen und zwar unbedingt über den Client! Sonst fehlt das lokale Wissen über die erfolgte Synchronisation. Bei weiteren PCs **löschen** Sie zunächst dort etwaige Duplikate der Daten und lassen Sie dann den dortigen Client die Daten vom Server herunterladen – sonst werden nur lokale Daten überschrieben, was länger dauert und oft zu „Stolperer“ führt wenn es die kleinste Abweichung im Datenbestand am Zweit-PC gibt.

Auf den meisten obigen Punkten wird im Folgenden weiter eingegangen.

### **2.2.1 Dateien: Browser / WebDAV / Synchronisierung Lokal <=> Server**

Mit ownCloud können Sie auf Ihre Datenbestände am Server auf drei grundverschiedenen Weisen vom eigenen PC / Notebook / Tablet aus zugreifen:

1. ownCloud bietet Benutzern eine recht aufgeräumte und gut nutzbare Oberfläche durch Login im Browser. Es besticht durch Baumansicht oder Dokumentenansicht (alle Ihrer Dokumente, unabhängig von ihren Verzeichnissen), und bietet zudem eine Sicht für Bilder sowie für alle chronologischen Aktivitäten. Das sind recht nützlich Features – doch wenn Sie Dateien haben, die Sie mit etwas Anderem als LibreOffice bearbeiten müssen, dann ist das hier nicht (direkt) möglich.
2. Als virtuelles Laufwerk mittels WebDAV einbinden.  
Ihre Daten erscheinen dann als ob ihr Rechner ein zusätzliches Laufwerk hätte. Damit können Sie alle Programme und Verfahren auf sie anwenden, die Sie sonst auch auf lokalen Dateien anwenden können.
3. Sie halten Ihre Daten lokal **und** am Server, und lassen Änderungen vom own-Cloud Client (ein Programm, das Sie auf Ihren Rechner installieren müssen) lau-

send synchronisieren. Das heißt: Änderungen, die Sie lokal vornehmen werden sofort an den Server übertragen; und Änderungen, die auf anderem Wege den Server erreichen, werden sofort auf Ihren Rechner übertragen.

Die Synchronisierung hat den großen Vorteil, dass Sie auch dann arbeiten können, wenn keine Internet-Verbindung besteht – z.B. am Laptop im Zug (oder eben auch wenn Ihre Internet-Verbindung mal streikt).

Die Arbeit direkt auf einem per WebDAV angebotenen Laufwerk hat „Eigenheiten“:

- Löschen einer Datei ist dann „unwiderruflich“ statt in den Papierkorb zu wandern; aber keine Angst – ownCloud hält auch gelöschte Dateien am Server vor, solange er genügend Plattenplatz dafür zur Verfügung hat.
- Gerade bei größeren Dateien kann z.B. eine Bearbeitung schleppend gehen wenn die Internet-Verbindung nicht sehr schnell ist.
- Wenn Sie zur Anbindung unter Windows das allgemein empfohlene Programm [Net-Drive 1](#) (FreeWare) benutzen, dann wird der Laufwerksbuchstabe für alle Benutzer des PCs sichtbar (wenn mehrere Benutzer sich gleichzeitig einloggen).

Wenn Sie sowohl NetDrive als auch ownCloud Client am gleichen PC für den gleichen Datenbestand aktivieren, dann werden Sie beim Speichern von LibreOffice-Dokumenten angeblökt, dass die Version am Server neuer ist als die ursprüngliche Fassung, ob sie dennoch überschrieben werden soll.

#### **Fazit:**

1. Setzen Sie **nie** die beiden Anbindungsmethoden **WebDAV + ownCloud-Client** für den gleichen Datenbestand am gleichen Client-PC ein.
2. Wenn Sie lokal über ausreichend Plattenplatz verfügen, so ist die **Synchronisierung** via ownCloud-Client dem Vorzug zu geben.
3. Andernfalls – vor allem an Tablets und Smartphones – binden Sie Ihre Daten als WebDAV-Laufwerk ein. Für Mobilgeräte gibt es auch ein ownCloud App.

### **2.2.2 Benutzer und Gruppen in ownCloud**

In früheren Versionen von ownCloud brachten Gruppen noch wenig Möglichkeiten mit sich. In ownCloud 6 ist die Unterstützung besser:

- Ein Benutzer kann ein Ordner oder eine Datei in seiner Sicht mit beliebigen anderen Benutzern oder Gruppen teilen. Dies geschieht in der „Dateien“ Ansicht von ownCloud (im Browser), in der Zeile des jew. Objekts (sichtbar erst bei Maus-Over).
- Der Administrator kann entscheiden, ob Benutzern erlaubt wird, mit Ihnen geteilten Inhalten erneut zu teilen: <Admin einloggen> | Administrator | Teilen
- Nachteil ist dabei, dass auf dieser Weise geteilten Inhalten bei den „Beschenkten“ nicht direkt als Ordner auf der obersten Ebene erscheinen, sondern unterhalb des Pseudo-Ordners „Shared“.
- Klickt der Mitnutzer zuerst auf „Shared“ dann sieht er alle für ihn erreichbaren „Shares“, doch steht oben in den Breadcrumbs „Du besitzt hier keine Berechtigung“.

gung, um Dateien hochzuladen oder zu erstellen“. Das ändert sich zwar, wenn er eine Ebene tiefer geht, ist aber zunächst irritierend.

- Über Gruppen können auch Kontakte und Kalender geteilt werden. Das ownCloud User Manual ist hierzu ausreichend verständlich.

### **2.2.3 Andere Datenquellen in ownCloud einhängen („Mounten“)**

ownCloud bietet die Möglichkeit, außerhalb von ownCloud gehaltene Daten einzubinden, als wären sie direkt im Verzeichnisraum eines ownCloud-Benutzers. Die Liste der bedienten Datenhaltungsarten ist beeindruckend:

***Lokale Plattenbereiche / Amazon S3 / Dropbox / FTP / Google Drive / Open-Stack Object Storage / ownCloud via WebDAV / SFTP / SMB-CIFS (Samba) (benötigt smbclient) / iRODS***

Das Gute daran ist – außer der großen Bandbreite an Möglichkeiten – dass für jedes Share ein „Ordnername“ angegeben wird, unter dem der externe Speicher bei jedem Benutzer mit Zugriffsrecht darauf auf **oberster Ebene** eingebunden wird. Das ist in der Handhabung (per Browser und WebDAV) angenehmer als bei über Gruppen geteilten Ordnern, die erst im Pseudo-Ordner „Shared“ auftauchen (s. oben).

Solche Shares können einer beliebigen Liste von Einzelbenutzern und Gruppen zugänglich gemacht werden, so dass der gemeinsame Zugriff möglich ist. Dies ist besonders wichtig angesichts der Einschränkung des Clients, der an einem PC immer nur ein Benutzerkonto synchronisieren kann. Ein solches Share, der im Sichtraum eines Benutzers liegt, kann dann auch vom ownCloud-Client synchronisiert werden.

Der Administrator kann es sich selbst vorbehalten, solche Anbindungen zu definieren; oder er kann Benutzern dies auch erlauben – sie finden es dann unter {Kennung (oben rechts)} | Persönlich | Externer Speicher

Der Administrator verwaltet „Shares“ und die Aktivierung für Benutzer unter {Kennung (oben rechts)} | Administration | Externer Speicher

#### **2.2.3.1 Lokale Plattenbereiche – local external storage**

**Das Anbinden von Plattenbereichen auf dem ownCloud-Server außerhalb von ownCloud ist aus Sicherheitsgründen dem Administrator vorbehalten!**

So wird's gemacht:

- Wenn das Server-Verzeichnis noch nicht existiert, z.B. weil Sie noch die Dateien (über ownCloud) hochladen müssen:
  - cd /var/www/vhosts/<IhrDomain>
  - mkdir external external/<Ordnername>
  - chown <DomainFTPUser>:psacln external external/<Ordnername>
- Mit Administrator-Kennung einloggen (Local Storage wird nur ihm angeboten)
- (Unten links) "+ Apps" | Ggf. "External Storage Support" aktivieren

- (Oben rechts) "IhrKonto" | Administration | Externer Speicher
  - Speicher hinzufügen | Local
  - Ordername vergeben  
(unter dem der lokale Pfad in ownCloud bereitgestellt wird)
  - Konfiguration => Lokaler Pfad am Server hier eingeben
  - Zutreffend => Hier Benutzer / Gruppen auswählen, die es sehen sollen

Man kann übrigens die gleiche Wirkung ownCloud in einer SSH-Sitzung **unterjubeln**:

- `cd /<IhrPfadZu>/owncloud/data/<IhrKontoName>/files`
- `In --symbolic <Ordnername> <Lokaler Pfad am Server>`

### **„Offizielle“ Local Storage oder „unterjubeln“?**

Die offizielle Methode mit Local Storage wäre leichter für Ihren Vertreter nachzuvollziehen wenn Sie z.B. in Urlaub sind. Der SymLink liegt m.E. dennoch leicht im Vorteil – immerhin spart es die Bindeglied-Software „Local Storage“.

In beiden Fällen gilt: <Lokaler Pfad am Server> muss nun in `open_basedir` von PHP aufgenommen werden (wenn Sie `open_basedir` nicht ausschalten)! In Plesk wie folgt:

Plesk | <IhrDomain> | PHP Settings |

`open_basedir = "<Lokaler Pfad am Server>:<Bisherige Einträge>"`

## 2.2.3.2 Wie ownCloud Änderungen an Fremdbeständen (nicht) merkt

Egal ob Dateien im User-Space liegen, oder per external Storage oder einer anderen Methode eingebunden werden: ownCloud hat ein grundsätzliches Problem, wenn sie von anderer Seite (eine andere Instanz bzw. Anwendung) geändert werden, so dass er die Änderung nicht direkt in seinen DB-Tabellen notieren kann. Das Problem: Wieviel Aufwand soll er vorsorglich treiben, um mögliche Änderungen zu entdecken?

Ist er dabei übereifrig, so läuft er Gefahr, eine ungewollt hohe Rechnerbelastung bzw. Antwortzeit zu verursachen; macht er wiederum zu wenig dafür läuft er Gefahr, aus den veralteten Informationen in seinen DB-Tabellen eine veraltete Sicht der Daten zu präsentieren. Das Zauberwort hier heißt „scan“ bzw. „rescan“.

Momentan führt ownCloud beim Betreten eines Verzeichnisses (durch bel. Benutzeraktion) einen sog. „shallow scan“ (flache Inspektion) durch. Genauer: Wenn die `mtime` (modification time – Änderungszeit) eines Verzeichnisses geändert ist, führt er den Scan auf den Inhalt des Verzeichnisses durch.

Dieses Verfahren klingt ausgewogen was die Belastung betrifft, hat aber in der Praxis Schwächen. Das Verzeichnis selbst wird in der UI mit der alten Zeit angezeigt, erst beim Betreten werden die neuen Inhalte angezeigt. Kamen veränderte Inhalte in ein Verzeichnis durch z.B. Überschreiben einer vorhandenen Datei, so wird sein `mtime` nicht geändert und folglich wird nicht einmal der „shallow scan“ ausgeführt.



Das ist in vielen Szenarien unzureichend, für die tägliche Benutzung unbrauchbar. In manchen Szenarien, z.B. eigene Backup-Skripts, kann man durch einen „touch“ auf verwendeten Verzeichnissen ein brauchbares Ergebnis erreichen. Ansonsten kann ownCloud veranlasst werden, für einen oder alle Benutzer seine Indexe komplett neu aufzubauen, mit einem PHP-Job am Server:

- `php /var/www/vhosts/<Ihr Domain>/console.php files:scan <User-Name>`

Diesen CLI-Job (Command Line Interface) kann man in passenden Szenarien per Cron regelmäßig ausführen lassen. Legen Sie zunächst einen Bash-Script an:

ownscanBackups.sh :

```
#!/bin/bash
php /var/www/vhosts/<Ihr Domain>/console.php files:scan <User-Name>
exit 0
```

und tragen Sie den Job in `/etc/crontab` ein (Zeile am Ende anhängen; dieses Beispiel lässt den Job täglich um 5:00 früh laufen):

```
0 5 * * * root [ -x /your/path/to/ownscanBackups.sh ] &&
/your/path/to/ownscanBackups.sh >/dev/null 2>&1
```

## 2.2.4 Der Client - gut, mit kleinen Einschränkungen

Den [Client](#) gibt es für Windows, Mac und Linux [hier](#). Lassen Sie sich nicht von der kostenpflichtigen „Enterprise Version“ irritieren – es ist die gleiche Software wie im Open-Source, nur mit Support.

Er synchronisiert ganze Dateibäume zwischen einem lokalen Rechner und dem Server. Allerdings (ohne zu tricksen) pro lokaler Rechner nur für **ein** ownCloud-Benutzerkonto. Gerade wenn Sie z.B. mehrere Datenbestände aus dem Privatleben, Beruf, Vereine oder Non-Profit Organisationen anbinden wollen – egal ob auf einem oder mehreren ownCloud-Servern – so wirkt sich diese Einschränkung empfindlich aus.

Die Möglichkeit, mehr als ein Benutzerkonto verwalten zu können, ist auf der Wunschliste der Entwickler, will aber aufgrund des Aufwands bisher nicht so recht passieren. Siehe dazu [hier](#) und [hier](#). Dafür gibt es mehrere Workarounds:

1. Die Daten des zweiten Kontos als "External Storage" am Server mounten, und im Client die verschiedenen Verzeichnissen zuordnen wie unten beschrieben.  
Vorteile: Völlig transparent an allen Clients, keine Zusatzarbeit am Client.  
Vorteil: Sichtbar auch im Web-Laufwerk, z.B. am Tablet / Smartphone.
2. Windows: Den Desktop-Link zu ownCloud manuell als anderer Benutzer starten: Shift-Taste halten und Rechtsklick => Als andere Benutzer ausführen.  
Das klappt weil die Config-Datei im User-Space lebt, also gibt es dann zwei, z.B. `C:\Users\Alice\AppData\Local\ownCloud` + `C:\Users\Bob\AppData\Local\ownCloud`  
Nachteile: Win-User anlegen, manuell zweiter Client starten.
3. Wie (2), aber als Job starten lassen beim Einloggen statt manuell starten:  
Start { | Alle Programme } | Verwaltung | Aufgabenplanung | Aufgabe erstellen  
Weiter siehe [hier](#) => `wolfpackmars2`

4. Mac: Siehe [hier](#).

**Fazit:** Benutzen Sie „Lokalen externen Speicher“ am Server, das geht problemlos, schnell und sauber (schnell in der Einrichtung UND schnell in der Benutzung).

Legen Sie dazu am Server manuell jeweils ein leeres Verzeichnis im ownCloud Wurzelverzeichnis an, in dem die Daten der Firma, des Vereins usw. liegen sollen. Stellen Sie die Verzeichnisse jeweils als [Custom Mounts](#) bereit, aber jeweils nur einer Gruppe verfügbar machen. Jeder Benutzer braucht dann nur die Gruppenmitgliedschaften zu erhalten, für die er berechtigt ist.

## **2.3 Benutzer und Gruppen anlegen**

Jetzt können Sie zur Tat schreiten, nachdem Sie eine Strategie entsprechend Ihren Bedürfnissen erarbeitet haben.

Die Oberfläche von ownCloud ist sehr aufgeräumt – so sehr dass man sich teils schwer tut, die Funktionalität zu finden, doch man gewöhnt sich schnell daran.

{Als Admin einloggen} | Benutzer (aus Dropdown-Menü oben rechts)

Wer ein aufgeblähtes Verwaltungsdialog erwartet, wird hier enttäuscht. Die Verwaltung von Benutzern und Gruppen besteht aus ganze zwei Elementen:

1. Eine Zeile zur Eingabe neuer Benutzer (und Gruppen!)  
Geben Sie Loginname und Passwort ein, Wählen Sie aus vorhandenen Gruppen und richten Sie neue Gruppen ein, alles in dem „Gruppen“ Dropdown. Klicken Sie auf „Erstellen“. Das war's schon.
2. Eine Liste der vorhandenen Benutzern, mit folgenden Feldern:  
{Avatar bzw. Bild} / Benutzername / Vollständiger Name / Passwort / Gruppen / Gruppenadministrator / Speicher

## **2.4 Dateien hochladen: Die Reihenfolge ist wichtig**

Vor allem, lokal am PC vorhandene Dateien, die synchronisiert werden sollen, NICHT manuell (z.B. per WebDAV) zu ownCloud hochladen, sondern vom Client hochladen lassen. Ansonsten holt sie der Client beim Anbinden alle nochmal (d.h. er überbuttert die lokalen Dateien mit identischen Kopien) und setzt das lokale Datum auf heute!

## **2.5 Den Client konfigurieren**

Das für Ihr Betriebssystem passende Programm [herunterladen](#) und installieren.

Beim ersten Aufruf wird der "ownCloud Verbindungsassistent" gestartet:

- Serveradresse: <https://<IhrDomain>/owncloud>
- Benutzername / PW: <IhrOwnCloudBenutzerName> / <IhrPasswort>
- Lokaler Ordner: **Wichtig!** Einen leeren Ordner an Ihrem PC wählen!
- Sobald Sie auf "Verbinden" klicken, wird mit dem Server verbunden und **alles**, was sich (auf welchem Wege immer – z.B. per WebDAV oder Anbindung lokaler Server-

Storage) wird synchronisiert, sprich heruntergeladen. Anders gesagt, es wird immer zunächst diese sog. „root“ Anbindung hergestellt, ob man will oder nicht.

- Falls das Benutzerkonto am Server nicht leer ist und ein Herunterladen nicht gewünscht ist, den ownCloud-Client aus der Taskbar sofort wieder anzeigen lassen und den Synchronisierungslauf **anhalten**. Die Zuordnung wieder "entfernen" (oben rechts), und schon heruntergeladenen Dateien am lokalen Rechner wieder löschen.
- Wichtig: Klicken Sie im Client unten rechts auf „Ignorierte Dateien bearbeiten“ und fügen Sie die Muster hinzu, die Sie über die Default-Liste hinaus nicht synchronisiert haben wollen. Bei mir waren das „\*.bak“ (von Ultra-Edit) sowie „temp/sess\_“ (PHP Session-Dateien im eigenen Ordner). Dies tun, BEVOR Sie solche Dateien hochladen, sonst sind sie am Server, können aber niemals dort vom Client gelöscht werden, da ignoriert – d.h. sie löschen sie lokal, und der Server will sie fortan herunterladen, da sie ja clientseitig fehlen... Mehr Info siehe Ignorierte Dateien / Sonderfall .htaccess.
- Jetzt können Sie im ownCloud Client nacheinander eine beliebige Anzahl Verbindungen zu lokalen Ordnern anfügen, genau wie Sie es brauchen, z.B. um gewohnte oder zweckmäßige Strukturen am PC zu erhalten, oder um bestimmte Teile der Benutzerdaten am Server eben nicht zu synchronisieren. Letzteres bietet sich an für Shares oder Datensicherungen, die Sie nur selten brauchen (die können Sie dann ggf. per Browser-GUI oder WebDAV einsehen).
  - Ordner hinzufügen
  - Wählen Sie... (den lokalen Ordner) (Alias wird übernommen, so lassen)
  - Weiter
  - Fern-Zielordner auswählen ODER  
Oben rechts "Ordner hinzufügen" und neuen (gleichnamigen) Ordner am Server anlegen
  - Dann unten rechts "Ordner hinzufügen" d.h. die Synchronisierung so festlegen

Die Übertragung der Dateien zum Server durch den Client erfolgt mit korrektem Zeichensatz der Dateinamen (UTF-8), auch von Windows aus; und es werden auch die Datümer der Dateien am Server korrekt gesetzt.

## **2.6 Weiterführende Themen**

### **2.6.1 NetDrive zur Kontrolle konfigurieren**

Wer die Sichtweise auf Dateien von Windows Explorer gewohnt ist – vor allem die Ansicht „Details“ – mag gelegentlich ein ownCloud-Datenbestand mittels NetDrive (per WebDAV-Protokoll) anbinden. Hier das Wichtigste dazu:

<https://gemeinwohl.info/owncloud/remote.php/webdav>

Advanced = UTF-8 + Https, Port = 443, <user> / <Passwort>

## 2.6.2 Andere Arten externer Speicher einhängen

Ich habe aus interessenshalber versucht ein anderes WebDAV-Laufwerk als externen Speicher einzuhängen, aber ohne Erfolg:

(Oben rechts) "IhrKonto" | Persönlich | Externer Speicher

Beispiel für "ownCloud / WebDAV"

- Ordernamen vergeben
- Konfiguration = WebDAV-Zugangsdaten

Es klappte aber nicht: `CURLOPT_FOLLOWLOCATION` cannot be activated when `safe_mode` is enabled or an `open_basedir` is set.

Das Problem ist bekannt, die Lösungen erfordern einen nicht-trivialen Eingriff in PHP-Code – ich habe erstmal die Bemühungen eingestellt.

Nachtrag: Inzwischen habe ich es geschafft, `open_basedir` auf Leer zu setzen – man muss dies durch eine entsprechende Anweisung in Plesks PHP Settings / Additional configuration directives: `open_basedir = ""`; ich vermute dass es damit gehen würde.

## 2.6.3 Tipp: Geteilte Dateien gleichzeitig bearbeiten

Mit ownCloud können Dateien von Benutzern geteilt werden, oder Shares können mehr als einem Benutzer zugeteilt werden. In beiden Fällen können Benutzer gleichzeitig im Browser-GUI die gleichen LibreOffice-Dateien bearbeiten. Hier zeigt sich eine Stärke von ownCloud: In einer rechten Spalte der Dokumentbearbeitung werden die Profilbilder der User gezeigt, die das Dokument gerade offen haben – und jeder erhält einen Schreibcursor in einer ihm zugeteilten Farbe. So kann jeder sehen, wo jeder anderer gerade arbeitet – eine spannende Möglichkeit zur kooperativen Dokumentbearbeitung!

## 2.6.4 Ignorierte Dateien / Sonderfall .htaccess

Der ownCloud-Client ignoriert Dateien, deren Namen einem Muster aus einer Liste von Mustern entspricht. Die Liste setzt sich aus einer globalen Liste zusammen plus etwaige Ergänzungen durch den Benutzer. Die globale Liste liegt im Installationsverzeichnis des Clients und kann nur dort editiert werden – ein Ein-/Ausschalten von Einträgen aus der Liste ist im Client selbst nicht möglich.

Darüber hinaus gibt es am ownCloud-Server eine „Blacklist“, die direkt im PHP-Code festgehalten ist. Dateien in der Blacklist werden nie übertragen – auch dann nicht, wenn ein Client dies machen will. Ein fester Eintrag im Code verbietet grundsätzlich die Übertragung von Dateien mit dem Namen „.htaccess“.

Dies wirkt zunächst evtl. befremdlich, doch man muss bedenken dass solche Dateien die Konfiguration des Apache-Webservers ergänzen, für alle Dateien im Verzeichnis wo sie liegen sowie alle Dateien darunter. Sie sind also mächtig und können leicht zu Sicherheitslücken oder Fehlfunktion führen. Zudem müssen sie sich oft leicht unterscheiden je nachdem, wie der Server am jew. Rechner insgesamt konfiguriert ist. Daher wäre eine direkte Übertragung oft nicht sinnvoll.

Workaround: Führen Sie je Server eine Variante der Datei z.B. `.htaccess.linux` + `.htaccess.windows`. Dann kopieren oder verlinken Sie am jew. Zielrechner die passende Variante der Datei.

In der Maske, wo man eigene Muster hinzufügen kann (das sind Linux-Shell Wildcard-Muster), steht bei jedem Muster eine Checkbox, und die Erklärung „Aktivierte Elemente werden ebenfalls gelöscht, wenn diese das Löschen eines Verzeichnisses verhindern. Dies ist nützlich für Metadaten.“ Gemeint ist, dass jedes sichtbare Muster greift, was die Synchronisierung von Dateien und Verzeichnisse betrifft; aber diejenigen, die ein Häkchen haben, werden auch gelöscht, wenn dadurch ein Verzeichnis gelöscht werden kann.

### **3. Eigene Kalenderdaten am Server halten mittels CalDAV**

Wer sich in die Denkweise der ownCloud-Autoren einfindet und mit der Bedienung vertraut wird, dürfte bei der CalDAV-Anbindung keine Probleme haben. Die [Beschreibung im User Manual](#) ist etwas dürftig, doch (fast) alles klappt intuitiv. Zwei Tipps:

- Import: Vergebens (noch) sucht man im Einstellungsdialog, das heruntergleitet wenn man oben rechts auf das Zahnrad klickt, nach einer Importmöglichkeit. Das geht (momentan) nur so, dass man eine exportierte iCalendar-Datei (.ics) zu den eigenen ownCloud-Dateien hochlädt und dort, in der ownCloud-Browser-GUI, unter Dateien zu ihr navigiert und darauf klickt!
- Android: DAVdroid erlaubt auch hier nur das Einbinden eines Kalenders auf einmal, wiederholen so oft nötig. CalDAV-Sync erlaubt es, alle Kalender der Kollektion auf einmal auszuwählen.

### **4. Eigene Kontaktdaten am Server halten mittels CardDAV**

DAS ist eine eher langwierige Geschichte mit vielen Stolpersteinen, die man umschiffen kann und muss - siehe mein Dokument:

Adressbuecher\_ownCloud\_Thunderbird\_Android\_<datum>.odt