OwnCloud

OwnCloud is a software i have been passively monitoring for some time now. evey once in a while i've installed it on my intranet server and tried out if it was ready for me to switch to it.

they have reached version 8 now (as of Mach 2015) and it has reached a state where i think i can use it as my primary file sharing solution internally in our household. It allowes me to store and manage my own data as well as my whifes data on our server at home. we can then acess it through the internet through a ssl encrypted link from anywhere and it also features a nice android app to keep some of your data even synced on your tablet and phone.

since I use linux and my whife uses windows, we appreciate the availability of a cross platform sync client. On my workstation at home i will mount my owncloud account through davfs2 which allowes me to access the “live” data. on my notebook and on my whifes notebook we are going to use the owncloud client which will synchronize the main part of our data. to access bulk data like pictures and stuff we can still use a webdav mount on either windows or linux which is available when online.

last but not least owncloud 8 features a fabulous web interface. it could still use some extra work in many places, but whatever feature is there works and whatever is missing is coming soon as they are under heavy development and release fast.

I think the project itself should be on pretty solid ground as it already has a huge user base throughout schools and universities.

So here are some things i have learned so far about using owncoud: By the way, i am running ubuntu on both my server and my desktops (xubuntu on the desktop side to be preceise) and my whifes notebook runs windows 7. we have android devices. whatever grows on trees is found only in our garden ;)

installation

that's very simple, just go to onwcloud.org and follow the installation instructions. for most distributions including ubuntu they hav a nice repo which can be added to your apt-sources. by using their own repo you make sure you are always running the latest security patches which is important for an application like this!

php-apcu

there seem to be some issues with the version of php-apcu that ubuntu 14.04 is currently using. owncloud 8 makes heavy use of it and therefore crashes about every hour or so when copying files because of that. the only thing that helps is to restart apache. a fix for this will soon be in the offical owncloud installation as they have added a version check and will only use php-apcu in versions they consider stable. until then, the easiest fix is to simply remove it and when you install new stuff, make sure it's not installed again on your system. this is only a temporary workaround until the version check is in their debian package

davfs2 mounts

locks

once you have installed davs2, make sure you disable the file locking mechanism. you can do this by editing `/etc/davfs2/davfs2.conf` and seting `use_locks 0` if you don't do this, all files created will just be empty hulls with no data inside

fstab

here's a line how you can use it in `/etc/fstab` to automount a webdav directory locally:

```
https://yourserver/remote.php/webdav       /home/psuter/data       davfs
uid=psuter,gid=psuter   0       0
```

passwords

since you will need a password you might want to save this information for mount to automatically mount your share on boot. you can do this by editing the file `/etc/davfs2/secrets`

```
https://yourserver/remote.php/webdav       youruser  yourpassword
```

make sure you edit the file as root and make sure the file is set to mode 0600 so it is readable only to root. after all it contains your password in clear text which is bad enough, so we don't want anybody else but root to read that.

Migrate Data

in order to migrate my data, i mounted my owncloud account through webdav as described above and then rsynced my data from its previous directory to webdav. make sure you have enough free space to do that in your owncloud data directory. the main advantage of rsync vs. cp is, that rsync allows to resume your copy process. this saved me from going nuts when i did not know about the apcu bug mentioned above.

one important thing thouhg: owncloud will not retain file change dates. it will set the file creation and change date to the current time when you ran the rsync. therefore, time checks should be disabled in rsync in order to do your sync. further more, since you are syncing this only once (at least i did) in order to speed things up, it is good enought to just compare file sizes in order to figure out if a file has been fully copied or not when you resume your sync process. so here is the rsync command i've used to do my migraton:

```
 rsync -vrx --size-only /local/source/ /mnt/webdav/target/
```

it is not advisale to use this line if you plan to keep your local and webdav storage synced for a while

because it will not detect if a file is modified but the size remains the same. for exmaple, if you correct a typo in a text file by changing a letter, this won't recoginze that!

Re-Share as samba mount

some devices in my network, like my HP OfficeJet MFP device need samba shares. My OfficeJet can scan to network shares, but they need to be windows shares. in order for such devices in my local network, i have created a user account like “local” to whom i can share directories i want my devices to access. so i have a “Scanned Documents” folder which is shared to “local”. on the server i then set up a local webdav mount for that user to say `/mnt/local` and share that mount via samba. this works perfectly and now my scanner can scan right into my owncloud directory which i then share with my whife. isn't that great :)

Android Webdav Clients

there seems to be a problem with many Android WebDAV clients when using mutliple Apache VirtualHosts on the same IP while using SSL encryption. FolderSync for example always reports “Not Found” when testing a connection.

In my Apache logs i found that the request for /remote.php/webdav went to my default VirtualHost rather than to the one i used for owncloud, so it seems that after checking the SSL cert there is no header information anymore containing the FQDN so that it can be used to redirect the reuqest to the right virtualhost.

some symlinking fiext the problem:
Here is the situation.
`DocumentRoot` of my owncloud installation: `/var/www/owncloud`
`DocumentRoot` of my default virtual host: `/var/www/pub`
URL of owncloud: `https://owncloud.mydomain.com/`

i then created a symlink in the default DocumentRoot to link to the owncloud root:

```
ln -s /var/www/owncloud /var/www/pub/owncloud
```

and a symlink within owncloud back to itself, so that even with the correct http header it will reach itself through the same url:

```
ln -s /var/www/ownlcoud /var/www/owncloud/owncloud
```

now i simply use `https://owncloud.mydomain.com/owncloud/remote.php/webdav` as my server's url and it all works :)

Windows WebDAV Mounts

The owncloud client for windows syncs your owncloud files to your local harddisk. however, you might

not want to sync everything as your notebook might not have enough space on its harddisk to keep all your files and files that where shared with you. If you still want the comfort of accessing files as if they where stored locally, you need to mount them as a webdav network share.

there is a webdav client in windows but it is totally messed up: - it “sometimes” does not support ssl - it does not support basic auth - can be fixed with a registry hack - it might still have connection issues despite the registry hack, in that case you need to install a hotfix - even after that i could not connect - according to severeal other users it also has issues with large files

that's why i've decided to follow the suggestion in the owncloud documentation and not use the native client. instead i found a tool called Net Drive that connects to all kinds of file sharing protocols and services and maps it as a network drive. it comes with a 30 day trial and seems to even be free after that with limited functionality. however, if you only need it to mount your owncloud share, the free version should do, as one of the limitations is that you can only use one share. the second limitation is, that it won't auto connect.. that, for me, is a reason to spend the 40$ once i am conviced the tools works. testing it now.

Fix Gallery (Switch to Gallery Plus)

Gallery in Owncloud 8 is broken. It may get stuck on certain images and keep loading forever. The recommended fix for this is, to move to Gallery Plus, which will eventually replace the current Gallery app as the stock Image and Video viewer app.

In order to support more filetypes, galleryplus uses imagemagick, wich must be installed first:

```
apt-get install imagemagick libmagickwand-dev
```

also make sure pear is installed:

```
apt-get install php-pear
```

and now install the pecl package for imagemagick:

```
pecl install imagick
```

now add this to your php ini file (if the pecl installer can't do it on its own, it will tell you to do it manually, was the case on my install)

```
 extension=imagick.so
```

now don't forget to reload apache

Now go to your `owncloud/apps/` directory and get the latest version from the git repo:

```
git clone -b stable8 https://github.com/interfasys/galleryplus.git
```

change file owner to www-data (so that the updater app will be happy)

```
chown -R www-data.www-data galleryplus
```

To **update** later on, use:

```
git pull --rebase origin stable8
```

you might need to run the chown from above again.. if you don't the updater app will probably tell you nex time you run an update :)

now to massively improve the performance, you need to apply those two patches at the description on the github page .. on my installation with OC 8.0.2 the second patch failed and i had to manually apply it.. now this really improves performance once you return to a previously loaded gallery page as it finally caches pictures!

JavaScript and CSS Asset Management

probably not so important if you don't have too much traffic on your server.. however i did it anyway :)

see the documentation

OCR for PDF files using ABBYY Fine Reader Corporate

Note you will need the corporate edition for this as this solution here is using hot folders which is only available in the corporate edition.

Note this is a work in progress.. i am not done yet with this setup!

I am seting up a windows vm in KVM on headless ubuntu server to run abby fine reader on my owncloud server.

i created a user “ocr” on owncloud that i used on my server to mount this users folder through davfs and then share it to my windows machine through samba. i then shared a folder that i will use as hotfolder with the user “ocr” and configured ABBY Fine Reader Hot Folders to scan this folder. i then created a second folder called “Originals” and a third called “Done”

The idea is: the scanner creates pdf's and saves them into the hot folder. on my HP MFP i can choose to scan to a samba share and i can set the basename for the pdf it generates.. like this i will have a way to later identify where to put the OCR'd file through a bash or php script running as a cron job on my owncloud server.. **Note** if you want owncloud's serch_lucene app to be able to index the pdf afterwards, use **PDF/A** as output format. files will be larger then with their other PDF format (actually about twice the size) but unfortunately search_lucene can't read those other pdf's.

one thing needs to be taken into account here: if finereader encounters any errors while saving the parsed file, it will just not save the output and move the input file to “originals”..

here is a one-liner which i might need to use in order to figure out which files where skipped when parsing, so i can copy them back to the hot folder to try again..

```
diff <(cd /mnt/owncloud/OCR/Done; find *) <(cd /mnt/owncloud/OCR/Originals;
```

```
find *)
```

synchronize external caldav and carddav sources

UPDATE I have moved away from python virtualenv as it broke vdirsyncer on almost every os update. I now run vdirsyncer in a docker container based on https://github.com/kaergel/docker-vdirsyncer. Basically copy the tempalte for the `docker-compose.yml` file and adjust to your needs. the `config` volume is a single file, so if you have multiple calendars to sync, put them all in one file and then pass this file as `/config` volume. I also added a directory which is mounted as volume to `/status` and used `/status` in the `general` section of my config file for the status directory. this seems to work fine.

caution this has only been done one-way, so that an external calender which is only changed externally gets regularly synced to owncloud. i haven't tired this in a two way mode.

prior to owncloud9 there was a sync script which did exactly that. unfortunately it no longer works with oc9 and the developer has posted that he won't fix it, so we need a new solution.

currently i am using vdirsyncer. to install in ubuntu 16.04 i did this:

```
apt-get install libxml2 libxml2-dev libxslt libxslt-dev zlib1g python
python-virtualenv python-pip
pip install pipsi
pipsi --home /opt/owncloud/ install --python python3 vdirsyncer
mkdir /opt/owncloud/vdirsyncer/status
chmod 777 /opt/owncloud/vdirsyncer/status
```

now it's time to configure your vdirsyncer.. in my case i wanted to sync a html read-only source (ical format) to owncloud. here is my config file:

```
[pair photokiste]
a = "photokiste_crm"
b = "photokiste_owncloud"
collections = null
conflict_resolution = "a wins"

[storage photokiste_crm]
type = "http"
url = "https://www.source-server.com/ical.ics"

[storage photokiste_owncloud]
type = "caldav"
url =
"https://my.owncloud.server/remote.php/dav/calendars/username/calenarname"
username = "myuser"
password = "mysecret"
```

the owncloud calendar name can be viewed by hovering over the calendar. it will appear as mouse pointer tooltip.

one thing on the ical format.. i used a library called PHP `iCalWriter` which i generated the ical source file with. as it turns out, PHP iCalWriter writes an different DTSTART and DTEND field for whole-day-events than what owncloud accepts. i have no idea who's write and who's wrong, but since i PHP iCalWriter is open source and it is already quite an old version i am using i decided to simply modify it. make sure, that your ical file contians a DTSTART and DTEND that look like this for whole-day-events (that is, events where the start and end date does not contain a time):

```
DTSTART;VALUE=DATE:20160903
```

and a line like this for date + time start and end dates:

```
DTSTART:19701025T030000
```

(PHP iCalWriter never added the ;VALUE=DATE: portion). this might be an onwcloud problem, as owncloud exports the “wrongly” specified events but it does not show them on the web-interface.. so maybe the web ui is just messing this up..

once the config file is written, vdirsyncer needs to be run to initialize the account. as the user who will run vdirsyncer later on run this command:

```
VDIRSYNCER_CONFIG=/opt/owncloud/photokiste.conf
/opt/owncloud/vdirsyncer/bin/vdirsyncer discover photokiste
```

now vdirsyncer can be run through cron or manually like this:

```
VDIRSYNCER_CONFIG=/opt/owncloud/photokiste.conf
/opt/owncloud/vdirsyncer/bin/vdirsyncer sync
```

that's it :) .. theoretically vdirsyncer should also support two way sync, but i haven't tried that.. however they seem confident in their solution so it might be worth a try even for two-way syncing :)

Update Vdirsyncer

you can run updates using pipsi as well:

```
pipsi --home /opt/owncloud/ upgrade
```

fix vdirsyncer after OS upgrade

you might just need to re-install vdirsyncer after a os upgrade because the python version behind the virtualenv changed and is no longer compatible with the environment. in that case just rename the `/opt/owncloud/vdirsyncer` directory and start from scratch as listed above. Make sure you keep your configs!

From:
http://wiki.psuter.ch/ - **pswiki**

Permanent link:
http://wiki.psuter.ch/doku.php?id=owncloud

Last update: **16.07.2021 00:30**