

# Seafile File Sharing

- [Installing Seafile with Docker](#)
  - [Seafile - File Sync and Drive On your Terms](#)

# Installing Seafire with Docker

# Seafile - File Sync and Drive On your Terms

<https://www.youtube.com/embed/gQ1WYgy6Z8s>

Seafile is a beautifully simple, and yet really powerful syncing and drive application that you can run on your own hardware, allowing you to maintain control of your data. It gets back to the simplest concepts of data syncing, and drive usage.

If you've been looking to ditch Google, Dropbox, Apple, and / or Microsoft for something less intrusive, and more private, then Seafile may just be the perfect fit for you, your family, your business, and more.

## Installation

### What you'll need

- Docker-CE and Docker-Compose installed
- (Optionally) Portainer or Portainer-CE Installed (Great GUI for Docker)
- Some level of Comfort / Familiarity with the Command line (for Linux)
- (Optionally) NGinX Proxy Manager

I have a video on getting [Docker-CE, Docker-Compose, Portainer, and NGinX Proxy Manager installed with 1 script, in under 5 minutes. Check it out.](#)

### Parts of this Application

- Server - A syncing server / centralized location
- Client - A machine or device you want to use for syncing data from and to.

- Syncing client - The client software that syncs specified libraries (folders) between your client machine(s) and the server.
- Drive client - client software that creates a mounted drive where you can drop data and have it selectively synced between your client machine(s) and the server.

## Warning: Sync and Backup are Very Different

To be clear, a sync action should never be used, or confused, as a backup of data. Backups will persist data even as data changes over time.

A sync will synchronize the data between various machines / devices. If you delete a file from one device, the sync action will then delete this file from all synchronized devices. The file is not backed up.

## Preparation for Install

On your server determine a directory where your synchronized data will be stored. I highly recommend keeping your docker installs in a well organized way. Check out my video on [Docker Organization](#) for some ideas.

Now, if you are a Portainer user ( a web GUI for docker, swarm, and kubernetes control ), then you can copy and paste the code block below into a new Stack on your Portainer instance, and deploy it -- making a few changes in the file of course.

If you're not a Portainer user, but are interested in a terrific graphical user interface for docker, then [check out my video on installing Portainer, and Portainer Agent](#).

If you prefer the terminal, then create a new directory called "seafile" wherever you keep all of your docker apps organized.

```
mkdir seafile
```

Now move into that directory

```
cd seafile
```

and create a new file called "docker-compose.yml":

```
nano docker-compose.yml
```

Copy the code block below, and paste it into the file you just created.

version: '2.0'

services:

db:

image: mariadb:10.5

container\_name: seafile-mysql

environment:

- MYSQL\_ROOT\_PASSWORD=<a really long strong password> # something like

E48flbicuqFagMBNX4508lke30s8d

- MYSQL\_LOG\_CONSOLE=true

volumes:

- /path/to/your/seafile/folder/or/data/store:/var/lib/mysql

networks:

- seafile-net

memcached:

image: memcached:1.5.6

container\_name: seafile-memcached

entrypoint: memcached -m 256

networks:

- seafile-net

seafile:

image: seafileltd/seafile-mc:latest

container\_name: seafile

ports:

- "8290:80" # you can change the left side to any open port

volumes:

- /path/to/your/seafile/data/storage/location/seafile-data:/shared # Requested, specifies the path to Seafile

data persistent store.

environment:

- DB\_HOST=db

- DB\_ROOT\_PASSWD=< same password as above >

- TIME\_ZONE=America/Chicago # Optional, but recommended

# - SEAFILE\_ADMIN\_EMAIL=me@example.com # I left this out in the video - check it out to understand why

# - SEAFILE\_ADMIN\_PASSWORD=asecret # I left this out in the video - check it out to understand why

depends\_on:

- db

- memcached

networks:

- seafile-net

```
networks:  
  seafile-net:
```

In the yaml code you just pasted, you need to change a few items. They are indicated by being surrounded with "<" and ">" symbols. Make sure you set a long, strong db password, it's not a password you need to know or have for later, just make sure the password you create is the same in both sections of the yaml code where it's indicated.

Next, you need to set the absolute path to the location where you'll store your data, and the location of the database. I suggest the database be stored in your normal docker organizational location. The data itself, may be stored on a separate drive, or location due to the size you'll want to reserve for file syncing. Set both paths properly.

Finally, you'll want to set the port on the left side of the colon ":" to a port that is open and available on your host system. This will be the port you'll use to access your Seafile web interface.

# Run our Seafile Container

Now we are ready to run the docker-compose command and get your server running.

```
docker-compose up -d
```

Once it has been pulled down, and you get the 'done' message in the terminal, use Portainer to check the logs, or you can do the command:

```
docker-compose logs -f seafile
```

You want to wait just a minute or two to make sure it's fully up and running.

Once you see an indicator that it is up, then go to your server IP and the port you set, in your browser.

I went to `http://192.168.10.26:8290` . You'll want to use your server IP and the port you set of course. But once, there, you'll login with the default username and password.

username: me@example.com

password: asecret

Now, go to the System Admin section, and create a new user.

Once created, wait just a couple of minutes for everything to complete on the creation of your user, then make your new user an Admin. You'll want to type the first letter of the username into the field, and you'll see the option show up in the drop list. Select it, and save.

Now, you can delete the default user. Log out, and then log in with your new user. You'll be prompted immediately to change the password you set for the new user, so use a temporary password when first setting up the new user. Change the password now to a long, strong password, and again, you'll be prompted to login. This time you'll be logged in as your new user, and you'll have full admin rights over the system.

Now you can start creating libraries, other users, user groups, and so on.

Finally, you'll want to go to the Seafile download pages, and get the Syncing client, and the Drive client for your various machines, and mobile devices, and get them connected up to your new install.

You can select which libraries should be synced for each machine / device. while it's an extra step, it's also super useful to not have every thing automatically sync to every device you add.

Check out the video for more on setting up and configuring the Sync and Drive clients.

# Support my Channel and Content

Support my Channel and ongoing efforts through Patreon:

<https://patreon.com/awesomeopensource>