Migrate Nextcloud To Another Server

Restoring from backup. Use this same method for migrating.

1) You use Rsync to move the backed up files to the new server. It will keep the tags and other permission that the backup was oriniglally stored with. If you don't care about that, just copy it. (rsync s pretty quick and copying either way).

2) I found it is easier to start a new instance of your new next cloud, then, copy over your data. This is because some files that are too old, dont sync well with the latest versions and, we are also getting rid of anything else that is not compatible... for whatever reasons. So... lets get started.

First things first. Where are you storing you volumes? I am putting them here:

`/home/server/nextcloud/`

I base everything off of the yml files. You can use hub.docker.com or you can create your own. I did a little of both here.

Create 2 new folders/directories in that same location for the 2 new volumes...
if i am in the `/home/server/nextcloud/` directory I would type:

```
sudo mkdir data datadb
```

Put your yml file in here `/home/server/nextcloud` or anywhere you want to keep that project file at.

I am going to copy over the database to the new server

```
rsync -Aax /backup/directory/nextcloud/datadb /home/server/nextcloud/datadb
```

I am going to NOT move the data files yet. We have a /data folder ready to go and empty for nextcloud to fill up.

Start up your new nextcloud now.
-Make sure it is running
-Make sure the site is up and it gives you the sign in screen.

Create a new login

All of your old info will be stored/saved however this requires you to create a new admin account. Later, you can remove it using one of your original user names.

While creating this, sometimes its slow. Sometimes its not.

> You may want to speed this up by going over to your php stuff.

It didnt work if:

If, during sign in, if you see it's asking for database info, it didnt work.

You dont need to wait for it to load. If it looks like it is starting to install stuff, you can leave. We just wanted it to have enough info to start creating the new config.php file.

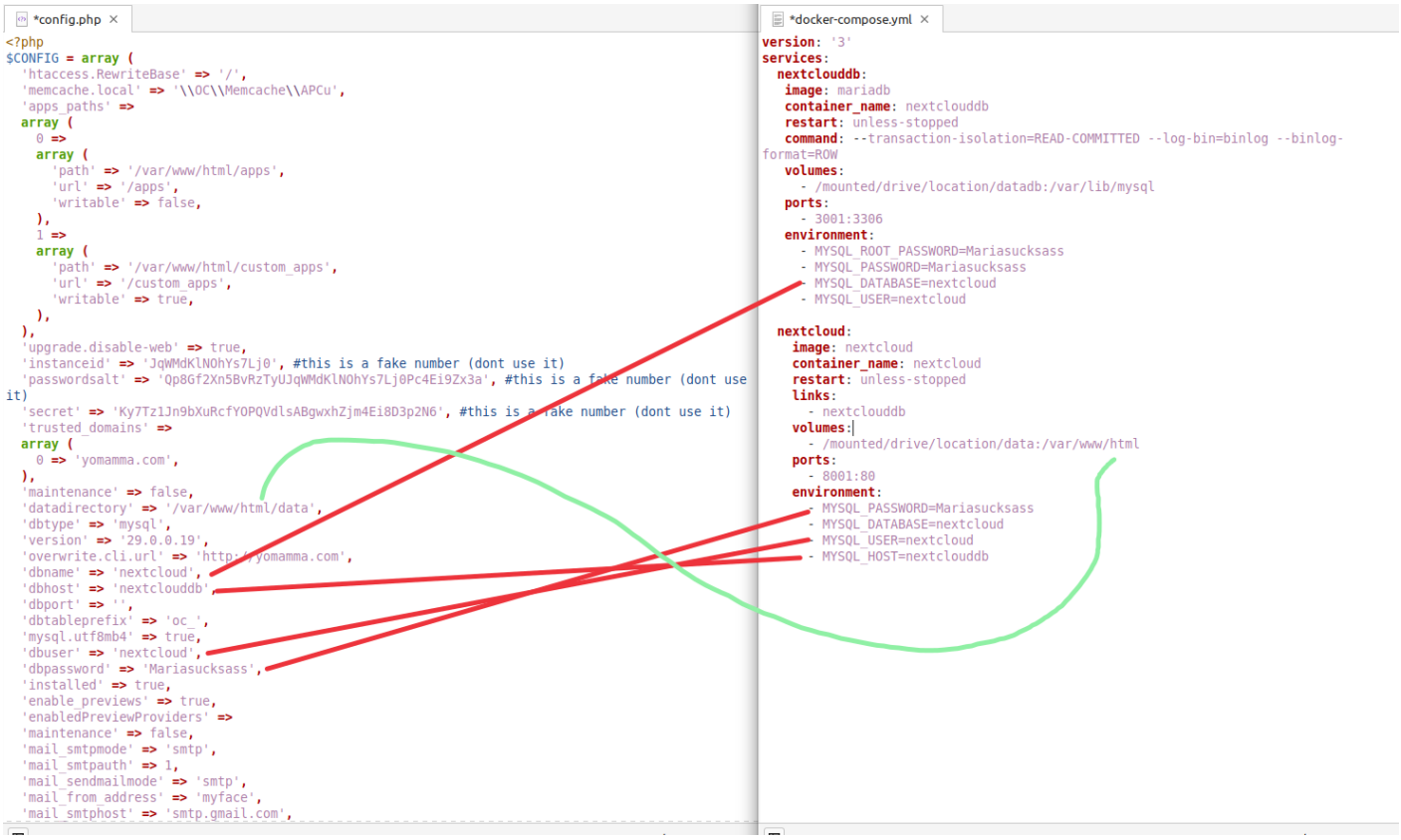
You can shut down your container.

Config file

So! Lets get into that config file and compare. Mine is located in `/home/server/nextcloud/data/config/`

I compare my .yaml file to the config.php, to make sure all my info is correct. Verify the other info not stated here like, trusted domains, overwrtie cli or anything else.

Note: the database info needs to match what is in your original config file so, if you need to, use that config file and copy the fields needed to make this new config file work.



Moving old data file

Now we can finally moving the main data in. Not all of it. ONLY the data folder that contains the users data. All files and folders that are before it, don't copy them over.

That means I'll be copying this one:

```
rsync -Aax /backup/directory/nextcloud/data/data/* /home/server/nextcloud/data/data
```

Notice the "data/data". We are deeper into the directory now. Here you should have all the files of all the users.

Start it up

If your new config file matches everything that is needed from your old one and, all your folders/files are copied over correctly, you should be good to go.

Start up a new container or... whatever.

If you came from an older Nextcloud version:

It might tell you to update using the terminal/command line. If so, try this.

If you dont need to update or, you finished the update, go to the next step.

Oh wait! there is no next step. You should only need to refresh your browser and it SHOULD work? Holy shit we did it!

This was my take on it. I just dont understand how nextcloud explains it. But...

If none of that works, try the below but I have not been able to get past step 5 using docker.

Setting up the New Server

1. **Install the Operating System**: Set up the new machine with the desired operating system.
2. **Configure Web Server and PHP**: Install and configure the web server (e.g., Apache or Nginx) and PHP for Nextcloud. Ensure that the PHP version matches Nextcloud's supported configuration, and all relevant PHP extensions are installed. Also, set the appropriate permissions and file upload size limits.
3. **Set up the Database**: Set up the database and ensure it's a Nextcloud-supported configuration.
4. **Copy Configuration (Optional)**: If your original machine was installed recently, you can safely copy the base configuration from the old server to the new server as a best practice.

Migrating Data from the Old Server

1. **Enable Maintenance Mode**: On the old server, stop Nextcloud and activate the maintenance mode. Wait for 6-7 minutes to allow all sync clients to register that the server is in maintenance mode.
    1. Note: if you do not have a running older system or have only files/folders of that system, there is no need to put it into maintenance mode as there is no front end client to pause.
2. **Stop Nextcloud**: After the waiting period, stop the application and/or web server that serves Nextcloud.
3. **Create a Database Backup**: Create a dump of the database on the old server and copy it to the new machine. Then, import the dump into the database on the new server. (See "Backup and Restoring backup" for more details.)
4. **Copy Nextcloud Files**: Copy all files from your Nextcloud instance on the old server, including the Nextcloud program files, data files, log files, and configuration files, to the new machine. Ensure that the data files retain their original timestamps (you can use

`rsync` with the `-t` option) to prevent clients from re-downloading all files after the migration. Note that the file locations may vary depending on the original installation method and operating system. On the new system, make sure to place the files in the appropriate locations. If you change any paths, adapt the paths in the `config.php` file accordingly. (See "Backup and Restoring backup" for more details.)

5. **Check Data Fingerprint**: Check the `config.php` file of the old system to see if it has the `data-fingerprint` set to a non-empty value. If so, run the `maintenance:data-fingerprint` command on the new system, similar to how it's required when performing a backup restoration. (See "Restoring backup" for details.)

Testing and Finalizing the Migration

1. **Test the New Server**: While Nextcloud is still in maintenance mode on the new server (confirm!), start the database, web server, and application server. Access the migrated Nextcloud instance in your web browser. Confirm that you see the maintenance mode notice, log entries are written by both the web server and Nextcloud, and no error messages occur. Then, take Nextcloud out of maintenance mode and repeat the testing. Log in as an admin and confirm that Nextcloud is functioning normally.
2. **Update DNS**: Once you've confirmed that the new server is working correctly, change the CNAME entry in the DNS to point your users to the new server's location.

Full article from Nextcloud:

https://docs.nextcloud.com/server/stable/admin_manual/maintenance/migrating.html

---

Revision #23
Created 2024-05-31 22:41:52 UTC by Danicus
Updated 2025-03-15 23:30:02 UTC by Danicus