Docker FAQs:



Backup Script für docker volumes

Eine nützliche Anleitung findest du hier

Busybox

BusyBox wird heutzutage immer beliebter, besonders unter Docker-Benutzern. Viele Docker-Images verwenden BusyBox, um Ihnen ein minimales Image zur Verfügung zu stellen. Einen interessanten Artikel findest du hier: Busybox

Collabora Online in Nextcloud

Hier eine Anleitung für Ubuntu 18.04 und Nextcloud 16,

Bei der Eingabe des neuen Collabora Containers ist die eigene Nextcloud Domain einzugeben.

```
docker run -t -d -p 127.0.0.1:9980:9980 -e 'domain= cloud \\. nextcloud \\. com ' –restart always -–cap-add MKNOD collabora/code
```

Bei der Konfiguration in Nextcloud unter **Collabora** ist neben der Office Adresse auch der Port **443** einzugeben.

Befinden sich die Nextcloud und Collabora Office hinter einem Proxy, so soll diese Anleitung heilfen (noch nicht getestet)

Collabora Container Updaten

Folgende Schritte sind durchzuführen:

1. `docker container ls`

listet die Container auf

1. `docker container stop CONTAINER ID`

stoppt den Container

1. `docker container rm CONTAINER ID`

löscht den Container

1. `docker pull collabora/code`

holt den neuen Container

1. `docker run -t -d -p 127.0.0.1:9980:9980 -e 'domain=cloud\\.nextcloud\\.com' -–restart always -–cap-add MKNOD collabora/code`

Cron Jobs in Docker erstellen

zum Ausprobieren hier eine Anleitung

Deinstallieren von Docker

```
sudo dpkg -l | grep -i docker
sudo apt-get purge -y docker-engine docker docker.io docker-ce
sudo apt-get autoremove -y --purge docker-engine docker docker.io docker-ce
sudo rm -rf /var/lib/docker /etc/docker
sudo rm /etc/apparmor.d/docker
sudo groupdel docker
sudo rm -rf /var/run/docker.sock
```

Details siehe hier

Dockervolumes leicht erklärt

https://www.ionos.de/community/server-cloud-infrastructure/docker/docker-container-volumes-verstehen-und-verwalten/

Welche php.ini wurde geladen:
docker exec -it <container-name> php -i | grep „Loaded Configuration File“

Prüfen welche php.ini geladen wurde

```
docker exec -it <container-name> php -i | grep "Loaded Configuration File"
```

Dockervolumes für Konfigigurationsdateien anlegen

geht über named volumes. Die named Volumes werden in **/var/lib/docker/DOCKERVOLUMENAME/_data** gespeichert. In diese kann die Konfigurationsdatei gespeichert werden. Beachte auch das docker-compose.yml **ToDo** : diesen Schritt automatisieren

Fehlermeldungen

- Failed to start docker.service: Unit docker.service is masked.

Abhilfe: `systemctl unmask docker.service systemctl unmask docker.socket systemctl start docker.service`

Kopieren eines Files vom Host System in einen Container und umgehrt

Kopieren in einen Container:

```
docker cp beispiel.txt mycontainer:/beispiel.txt
```

Kopieren aus einem Container

```
docker cp mycontainer:/beispiel.txt beispiel.txt
```

Mehrere Files kopiert man folgendermaßen (z.B. vom src Directory):

```
docker cp src/. mycontainer:/target
docker cp mycontainer:/src/. target
```

Genauers siehe die Beschreibung von docker cp

Logs und Docker

Logs in docker ansehen

Nano in Docker installieren

```
docker exec -it id_container bash
apt-get update
apt-get install nano
export TERM=xterm
```

PHP Variablen aus einen Docker Container auslesen

den Docker Container starten

```
docker exec -it <docker container name oder docker id> /bin/bash
```

danach im Container folgenden Befehl eingeben:

```
php -i
```

danach die Variable suchen

Restart Policies

Eine genaue Beschreibung findest du hier:

Für Eilige hier eine Übersichtstabelle:

Policy	Erklärung
always restart	
unless-stopped	
on-failure restart	

Speicherort von Docker Containern

Dockercontainer werden in `/var/lib/docker` gespeichert.

From:
http://wiki.waldhofer.at/ - **Wiki von Franz**

Permanent link:
http://wiki.waldhofer.at/doku.php?id=docker:faqs&rev=1741422869

Last update: **2025/03/08 09:34**