

# Inhaltsverzeichnis

**PaperlessNGX (VM/RPI)** ..... 3



# PaperlessNGX (VM/RPI)

## Voraussetzungen:

Ins Internet würde ich meine Dokumente nie freigeben, deshalb entfällt das hier. Es läuft zwar auf einem RPI3 aber doch sehr träge. Ich würde klar VMM oder Proxmox empfehlen und dort als eigene VM die sonst nix anderes tut. Wie immer habe ich mich für Docker entschieden, damit ich mit der Synology VMM einen Backup Host habe. Als erstes legen wir Klon von unserem Template an und starten diesen. Wir legen auf der Synology einen freigegeben Ordner an „Paperless“, diesen geben wir entweder auf einen Host oder auf unser Netzwerk per NFS frei. Nun legen wir in dem Verzeichnis 3 Ordner an: archiv, dokumente & scanner\_ablage.

## Proxmox:

Wir legen einen Klon unseres Templates an.

## Beides:

Wir melden uns per SSH auf der VM an.

Wir legen nun die Ordner fürs Mounten an.

```
sudo mkdir /archiv
sudo mkdir /dokumente
sudo mkdir /paperlessngx
sudo mkdir /scanner_ablage
```

Nun bearbeiten wir die fstab, 3 Einträge sind gleich der 4te Eintrag ist abhängig von VMM/RPI oder Proxmox.Ihr müsst natürlich wieder auf Eure Volume Nummer achten.

```
sudo nano /etc/fstab
```

dort tragen wir ein

```
192.168.2.10:/volume1/paperless/archiv /archiv nfs auto,rw,soft,users 0 0
192.168.2.10:/volume1/paperless/dokumente /dokumente nfs auto,rw,soft,users
0 0
192.168.2.10:/volume1/paperless/scanner_ablage /scanner_ablage nfs
auto,rw,soft,users 0 0
```

und als letzten Punkt

```
192.168.2.10:den_pfad_den_Ihr_für_die_ContainerDaten_nutzt /paperlessngx nfs
auto,rw,soft,users 0 0
z.b.
192.168.2.10:/volume2/debian_prod/paperlessngx /paperlessngx nfs
auto,rw,soft,users 0 0
```

Nun gehen wir auf die Portainer Website und erstellen einen neuen Stack

Name: paperless

```
version: "3.9"
services:
  redis:
    image: docker.io/library/redis:7
    container_name: PaperlessNGX-REDIS
    restart: unless-stopped
    volumes:
      - redisdata:/data

  db:
    image: docker.io/library/postgres:15
    container_name: PaperlessNGX-DB
    restart: unless-stopped
    volumes:
      - /paperlessngx/db:/var/lib/postgresql/data
    environment:
      POSTGRES_DB: paperless //kann geändert werden, dann aber auch unten
      POSTGRES_USER: paperless //kann geändert werden, dann aber auch unten
      POSTGRES_PASSWORD: paperless //kann geändert werden, dann aber auch unten

  webserver:
    image: ghcr.io/paperless-ngx/paperless-ngx:latest
    container_name: PaperlessNGX
    restart: unless-stopped
    depends_on:
      - db
      - redis
      - gotenberg
      - tika
    ports:
      - "8777:8000"
    healthcheck:
      test: ["CMD", "curl", "-fs", "-S", "--max-time", "2",
"http://localhost:8000"]
      interval: 30s
      timeout: 10s
      retries: 5
    volumes:
      - /paperlessngx/data:/usr/src/paperless/data
      - /paperlessngx/media:/usr/src/paperless/media
      - /dokumente:/usr/src/paperless/media/documents/originals
      - /archiv:/usr/src/paperless/media/documents/archive
      - /paperlessngx/export:/usr/src/paperless/export
      - /scanner_ablage:/usr/src/paperless/consume
    environment:
      PAPERLESS_REDIS: redis://redis:6379
      PAPERLESS_DBHOST: db
      USERMAP_UID: UID_LDAPUSER
```

```
USERMAP_GID: UID_LDAPUSER
PAPERLESS_TIME_ZONE: Europe/Berlin
PAPERLESS_ADMIN_USER: ADMIN_USER //bitte ändern
PAPERLESS_ADMIN_PASSWORD: ADMIN_PASSWORD //bitte ändern
PAPERLESS_OCR_LANGUAGE: deu+eng
PAPERLESS_CONSUMER_RECURSIVE: true
PAPERLESS_CONSUMER_SUBDIRS_AS_TAGS: true
PAPERLESS_CONSUMER_POLLING: 60
PAPERLESS_SECRET_KEY: // sollte man eintragen
PAPERLESS_TIKA_ENABLED: 1
PAPERLESS_TIKA_GOTENBERG_ENDPOINT:
http://gotenberg:3000/forms/libreoffice/convert#
PAPERLESS_TIKA_ENDPOINT: http://tika:9998

gotenberg:
  image: docker.io/gotenberg/gotenberg:7.8
  container_name: PaperlessNGX-GOTENBERG
  restart: unless-stopped
  ports:
    - 3000:3000
  command:
    - "gotenberg"
    - "--chromium-disable-routes=true"

tika:
  image: ghcr.io/paperless-ngx/tika:latest
  container_name: PaperlessNGX-TIKA
  ports:
    - 9998:9998
  restart: unless-stopped

volumes:
  redisdata:
```

statt, danach ist die Weboberfläche über

```
http://192.168.2.91:8777/
```

erreichbar

From:

<https://nas-wiki.hundacker.eu/> - **NAS-Wiki**

Permanent link:

<https://nas-wiki.hundacker.eu/doku.php/virtual/docker/paperless>

Last update: **15.08.2023 15:11**

