

4

Checklist - nouveau container Docker

4 étapes avant de cliquer sur "Deploy"

0

Avant tout

Repérer les volumes dans la documentation

Va sur la page Docker Hub du logiciel (ou son dépôt GitHub). Cherche la section "Volumes" ou "Mounts" : elle liste les chemins internes du container. Ce sont eux que tu vas brancher sur ton Beelink.

```
# Exemple Stirling PDF -- volumes internes listés sur Docker Hub
/usr/share/tesseract-ocr/5/tessdata -> tessdata
/configs                             -> configs
/logs                                 -> logs
```

Règle simple : un chemin interne du container = un sous-dossier à créer sur ton Beelink. Note-les avant de passer à l'étape 1.

1

Créer les dossiers sur le Beelink

Ouvre ton terminal et crée exactement les sous-dossiers identifiés à l'étape 0 :

```
mkdir -p /home/USER/docker/nom-du-logiciel/tessdata
mkdir -p /home/USER/docker/nom-du-logiciel/configs
mkdir -p /home/USER/docker/nom-du-logiciel/logs
```

2

Adapter le YAML dans Portainer

Dans l'éditeur de Stack, remplace chaque volume par le chemin absolu du sous-dossier correspondant (identifié à l'étape 0) :

```
# Chemin interne docker -> chemin absolu sur le Beelink
- /home/USER/docker/nom/tessdata:/usr/share/tesseract-ocr/5/tessdata
- /home/USER/docker/nom/configs:/configs
- /home/USER/docker/nom/logs:/logs
```

Pourquoi le chemin complet ? Portainer stocke ses fichiers de manière invisible. Le chemin absolu garantit à 100% l'emplacement réel sur le disque -- indispensable pour Rclone.

3

Vérifier les droits (si besoin)

Si le logiciel affiche "Permission Denied" dans les logs, Docker n'a pas accès au dossier. Tape :

```
sudo chown -R USER:USER /home/USER/docker/nom-du-logiciel
sudo chmod -R 775 /home/USER/docker/nom-du-logiciel
```

Cette commande s'applique à tout le dossier du logiciel d'un coup -- pas besoin de la répéter pour chaque sous-dossier.