

# Proxmox : utiliser une image Docker (format OCI)

## Ressources

- <https://www.virtualizationhowto.com/2025/11/complete-guide-to-proxmox-containers-in-2025-docker-vms-lxc-and-new-oci-support/>
- <https://github.com/containers/skopeo>

## Présentation

La version de Proxmox VE 9.1 permet l'importation et l'utilisation d'image au format OCI pour créer des conteneurs LXC.

OCI est une organisation visant à créer un standard pour unifier la façon de stocker les containers.

- [opencontainers.org/](https://opencontainers.org/)

Voici quelques registres qui permettent l'hébergement d'images au format OCI :

- Docker hub : [docker.io](https://hub.docker.com/) (<https://hub.docker.com/>)
- Github Container Registry : [ghcr.io](https://ghcr.io) (<https://oalahurikar.github.io/app-dev-blog/Github/Github-Container-Registry>)
- Red Hat Quay : [quay.io](https://www.projectquay.io/) (<https://www.projectquay.io/>)
- etc.

## En cas de problème de droits sur le stockage (solution non testée)

Problème possible lors de l'utilisation d'un NAS :

- La conversion OCI→LXC peut nécessiter créer/écrire des fichiers et ajuster des propriétaires UID/GID dans le rootfs. Si le NAS refuse ces opérations du fait de `root_squash`, Proxmox échoue avec "os error 13".
- le serveur NAS configure dans sa directive **export**, `root_squash` (mapping qui dégrade root) ⇒ Proxmox ne peut pas faire les `chown/chmod` nécessaires pour extraire les couches de l'image OCI entraînant une erreur OS 13

Solution : privilégiez dans la définition de l'export **/etc/exports**, les paramètres suivants (`norootsquash`):

```
/export/proxmox 192.168.x.0/24(rw, sync, no_root_squash, no_subtree_check)
```

Puis exécutez la commande

```
exportfs -ra.
```

Le **no\_root\_squash** est la clé pour que Proxmox (root) puisse lire/écrire et `chown` le contenu.

## Configuration à faire avec Proxmox

- créer un stockage de type Directory
  - menu Datacenter → Storage → Add → Directory
  - Paramètres :
    - **ID** : templates (définir le nom de votre choix)
    - **Directory** : `/var/lib/vz` (ou un autre chemin de votre choix comme `/mnt/templates` )
    - **Content** : cochez **CT Template**
  - Cliquez sur **Add**.

En configurant la valeur **Directory** avec `/var/lib/vz`, les **templates OCI** seront stockés dans le dossier `/var/lib/vz/template/cache`.

## Importer une image OCI

- menu Datacenter → Storage → local → CT Templates,
- cliquez sur **Pull from OCI Registry**,
- téléchargez l'image OCI après avoir indiqué sa **référence**, puis cliqué sur le bouton **Query Tags** pour lister les tags disponibles et choisi le **tag** voulu.

Exemple pour l'image Docker **debian** :

**Pull from OCI Registry**

Reference:  Query Tags

Tag:

File name:

Download

Exemple pour l'image Docker **rancher/nginx** :

**Pull from OCI Registry**

Reference:  Query Tags

Tag:

File name:

Download

Après téléchargement, les images sont disponibles dans le dossier au format tar :

Storage 'templates-OCI' on node 'siohyp3' Help

<b>Summary</b>	<span>Upload</span> <span>Download from URL</span> <span>Templates</span> <span>Pull from OCI Registry</span> <span>Remove</span> <span>S</span>												
<b>CT Templates</b>	<table border="1"><thead><tr><th>Name</th><th>Date</th><th>Format</th><th>Size</th></tr></thead><tbody><tr><td>debian_13.2.tar</td><td>2026-01-07 22:34:05</td><td>tar</td><td>49.30 MB</td></tr><tr><td>nginx_latest.tar</td><td>2026-01-07 22:33:41</td><td>tar</td><td>237.69 MB</td></tr></tbody></table>	Name	Date	Format	Size	debian_13.2.tar	2026-01-07 22:34:05	tar	49.30 MB	nginx_latest.tar	2026-01-07 22:33:41	tar	237.69 MB
Name	Date	Format	Size										
debian_13.2.tar	2026-01-07 22:34:05	tar	49.30 MB										
nginx_latest.tar	2026-01-07 22:33:41	tar	237.69 MB										
<b>Permissions</b>													

## Création d'un conteneur LXC basée sur une image OCI importée

From:  
[/ - Les cours du BTS SIO](#)

Permanent link:  
[/doku.php/reseau/cloud/proxmox/imageocieva?rev=1767821771](#)

Last update: 2026/01/07 22:36

