

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>
- https://raymii.org/s/tutorials/Finally_run_Docker_containers_natively_in_Proxmox_9.1.html

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/

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://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 de créer/écrire des fichiers et d'ajuster des propriétaires UID/GID dans le rootfs.
* Si le serveur NAS a configuré 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**, le paramètre **no_root_squash**.

```
/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.

Création/configuration d'un stockage dans Proxmox pour importer les images OCI

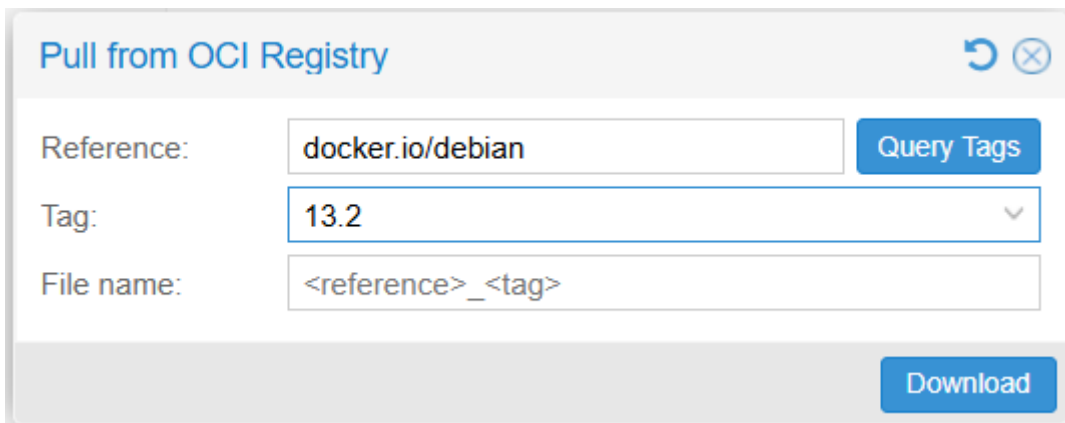
- Créer un stockage de type **Directory** :
 - Menu Datacenter → Storage → Add → Directory
 - Paramètres :
 - **ID** : templates (définir le nom de votre choix)
 - **Directory** : /var/lib/vz (ou un autre chemin de votre choix comme /mnt/templates)
 - **Content** : cochez **CT Template**
 - Cliquez sur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

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

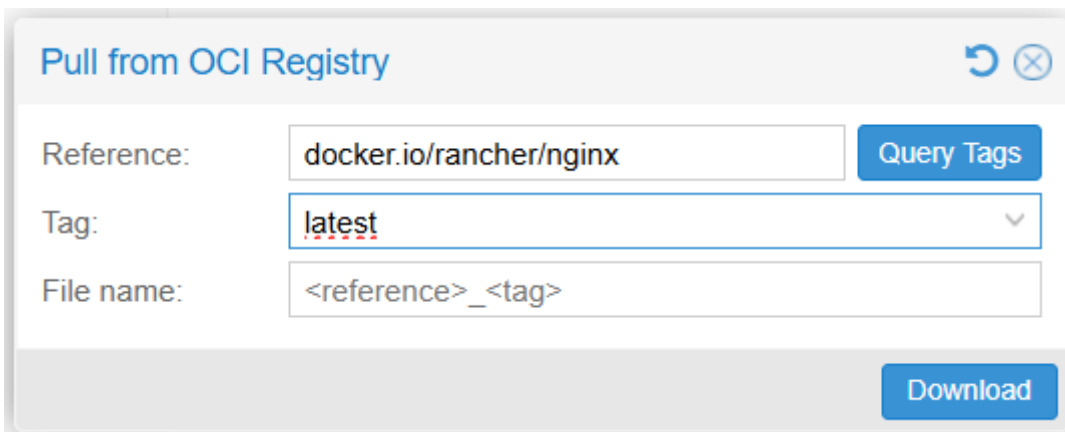
Importer une image OCI

- 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** :



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



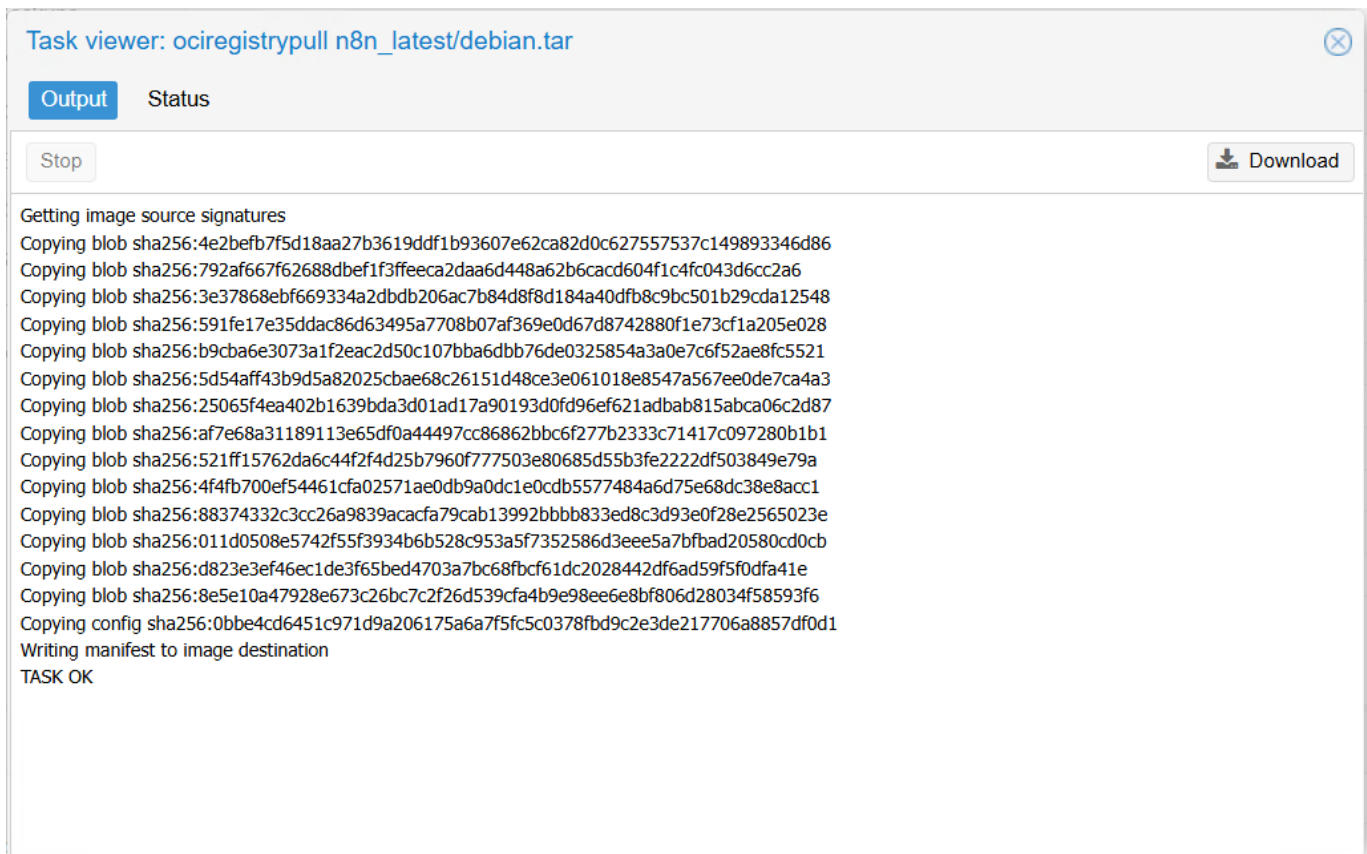
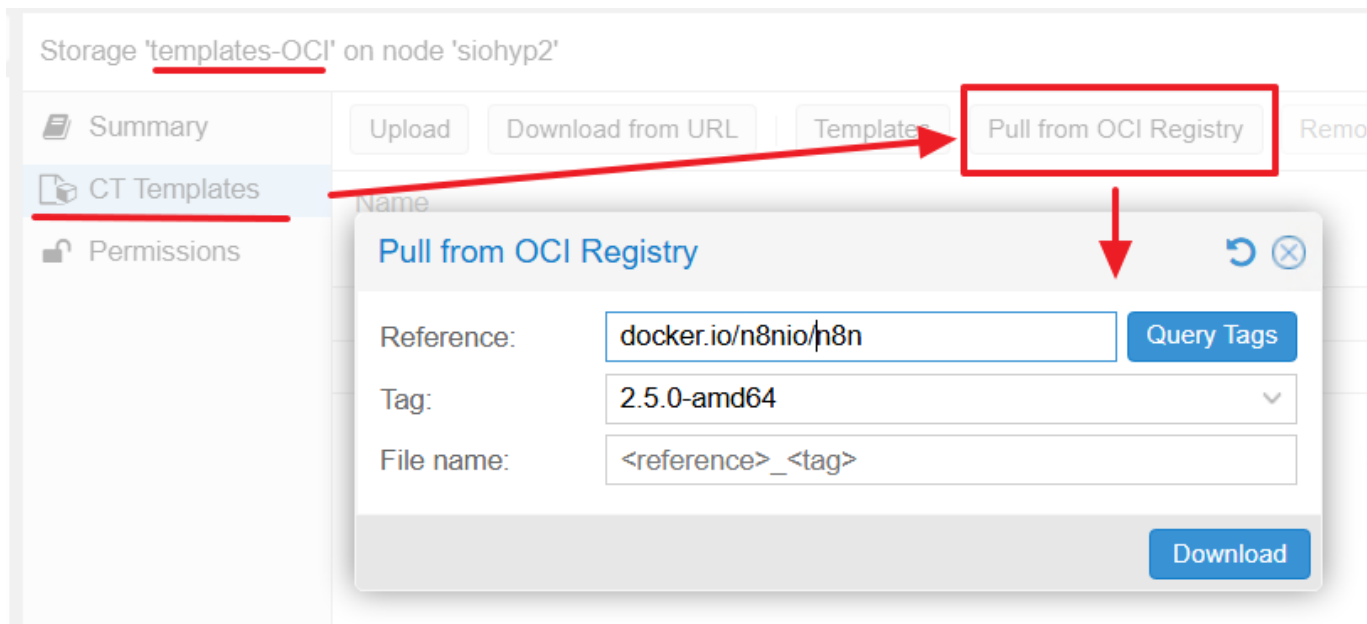
Après téléchargement, les images sont disponibles dans le dossier au format tar :

Storage 'templates-OCI' on node 'siohyp3'					Help	
	Upload	Download from URL	Templates	Pull from OCI Registry	Remove	S
CT Templates	Name	Date	Format	Size		
Permissions	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		

Exemple de création d'un conteneur LXC pour n8n

Importer l'image OCI n8nio/n8n:2.5.0

- 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.



Création du conteneur LXC

- Pour le choix du template, sélectionner comme stockage **template-OCI** puis le modèle n8n_2.5.0-amd64.tar :

Create: LXC Container

General **Template** Disks CPU Memory Network DNS Confirm

Storage: templates-OCI

Template: n8n_2.5.0-amd64.tar

[? Help](#) Advanced [Back](#) [Next](#)

- Pour le disque, rajoutez un volume **mp0** pour le mappage vers le dossier **/home/node/.n8n** du conteneur :

Create: LXC Container

General Template **Disks** CPU Memory Network DNS Confirm

rootfs	Mount Point ID:	0	Path:	/home/node/.n8n
mp0	Storage:	NFS-NAS	Backup:	<input checked="" type="checkbox"/>
	Disk size (GiB):	8		
	Enable quota:	<input type="checkbox"/>	ACLs:	Default
	Read-only:	<input type="checkbox"/>	Skip replication:	<input type="checkbox"/>
	Mount options:			

Advanced

From:
/ - Les cours du BTS SIO

Permanent link:
</doku.php/reseau/cloud/proxmox/imageocieva?rev=1769003538>

Last update: 2026/01/21 14:52

