

Synology NAS (NFS/ACL)

- [Résoudre les erreurs de permission NFS \(EACCES\) entre Synology et Docker](#)

Résoudre les erreurs de permission NFS (EACCES) entre Synology et Docker

❏ Problème d'ACL Synology avec NFS + Docker

“ **TL;DR** : Ton conteneur Docker ne peut pas écrire sur ton partage NFS Synology malgré des permissions `777` ? C'est l'ACL Windows qui prime sur les bits Unix.
Solution : `synoac1tool -del` récursif.

❏ Contexte

Tu as configuré un partage NFS sur ton Synology pour stocker les backups de Proxmox Backup Server (PBS) tournant dans Docker. PBS démarre, puis crash avec :

```
Error: unable to create rrdb stat dir - EACCES: Permission denied
```

Pourtant, `ls -la` affiche `drwxrwxrwx` et le propriétaire est bien `backup:backup` (UID 34).

La cause racine : Sur DSM, les dossiers partagés créés sur Btrfs/ext4 ont **l'ACL Windows activée par défaut**. Cette ACL prend le pas sur les permissions Unix classiques, même via NFS. Résultat : les bits `rw` visibles sont ignorés, et l'écriture est refusée pour l'UID du conteneur.

☐ Symptômes typiques

- Container Docker qui boucle en crash (`restart: always` + erreur `EACCES`).
 - `ls -la` affiche `drwxrwxrwx` ou `0777`, mais `mkdir` échoue.
 - Les lignes `ls -la` ont un `+` après les permissions : `drwxr-xr-x+`.
 - Le squash NFS est déjà sur "Pas de mappage" — donc ce n'est pas ça.
-

☐ Diagnostic rapide

1. Identifier l'UID attendu dans le conteneur

```
docker run --rm --entrypoint sh IMAGE -c "id backup"
# Exemple : uid=34(backup) gid=34(backup)
```

2. Vérifier les permissions actuelles sur l'hôte

```
ls -la /mnt/ton-partage-nfs
# Cherche le + après les permissions
```

3. Tester l'écriture réelle avec le bon UID

```
docker run --rm -v /mnt/ton-partage-nfs:/data --user 34:34 --entrypoint sh IMAGE \
-c "mkdir -p /data/test && echo OK || echo FAIL"
```

Si `FAIL` malgré `777` → **ACL Synology en cause.**

4. Confirmer via SSH sur le Synology

```
ssh admin@IP_SYNOLOGY
sudo synoacldtool -get /volume1/TonDossier
# Si tu vois des entrées [0], [1], etc. → ACL active
```

□ Résolution (étape par étape)

Étape 1 : Supprimer l'ACL du dossier racine

```
sudo synoacldtool -del /volume1/TonDossier
```

Étape 2 : Retirer les flags d'archive ACL

```
sudo synoacldtool -del-archive /volume1/TonDossier has_ACL
sudo synoacldtool -del-archive /volume1/TonDossier is_support_ACL
```

Étape 3 : Nettoyer récursivement toutes les ACL des sous-éléments

```
sudo find /volume1/TonDossier -exec synoacldtool -del {} \;
```

Étape 4 : Vérifier qu'il n'y a plus de +

```
ls -la /volume1/TonDossier
# Plus de + après les permissions (sauf @eaDir, dossier système)
```

Étape 5 : Remettre des permissions Unix propres

```
chown -R 34:34 /volume1/TonDossier
chmod -R 0755 /volume1/TonDossier
```

Étape 6 : Retester depuis l'hôte Docker

```
docker run --rm -v /mnt/ton-partage-nfs:/data --user 34:34 --entrypoint sh IMAGE \
-c "mkdir -p /data/test && echo OK || echo FAIL"
```

Si `OK` → `✓` C'est résolu.

☐ Bonnes pratiques préventives

Pour tes futurs partages NFS Docker :

Réglage	Valeur recommandée
Squash	Pas de mappage (No mapping)
Sécurité	<code>sys</code> uniquement (décoche Kerberos si pas utilisé)
Permissions	Appliquer <code>synoacftool -del</code> dès la création si tu utilises Docker

☐ Référence rapide (copier-coller)

```
# SSH sur Synology
ssh admin@IP_SYNOLOGY

# Supprimer ACL
sudo synoacftool -del /volume1/TonDossier
sudo synoacftool -del-archive /volume1/TonDossier has_ACL
sudo synoacftool -del-archive /volume1/TonDossier is_support_ACL
sudo find /volume1/TonDossier -exec synoacftool -del {} \;

# Vérification
ls -la /volume1/TonDossier
```

```
# Côté hôte Docker

chown -R 34:34 /mnt/ton-partage-nfs

chmod -R 0755 /mnt/ton-partage-nfs


# Test

docker run --rm -v /mnt/ton-partage-nfs:/data --user 34:34 --entrypoint sh IMAGE \
  -c "mkdir -p /data/test && echo OK || echo FAIL"
```

📁 Le saviez-vous ?

“ Les ACL (Access Control Lists) sont un héritage du monde Windows (NTFS). Synology les a intégrées pour faciliter la compatibilité SMB, mais elles créent des conflits silencieux avec les workflows Linux/NFS/Docker qui s'attendent à des permissions Unix classiques. C'est un peu comme essayer de conduire une voiture avec un volant à droite sur une autoroute française : techniquement possible, mais tout le système est conçu pour l'autre sens.

Auteur : Jonathan BIODORE

Date : 27 septembre 2026

Tags : #Synology #NFS #Docker #Proxmox #ACL #Troubleshooting