

Nextcloud

Comandi Docker per Nextcloud (occ)

“ **Premessa** I comandi `occ` vanno **sempre** eseguiti come utente `www-data`. Sostituisci `CONTAINER` con il nome del tuo container Nextcloud (lo trovi con `docker ps`).

occ — comando base

```
docker exec -u www-data CONTAINER php occ <comando>
docker exec -u www-data CONTAINER php occ list # elenco di TUTTI i comandi
```

Manutenzione

```
# Modalità manutenzione (attiva prima di interventi pesanti / backup)
docker exec -u www-data CONTAINER php occ maintenance:mode --on
docker exec -u www-data CONTAINER php occ maintenance:mode --off

# Riparazioni dopo aggiornamenti o problemi
docker exec -u www-data CONTAINER php occ maintenance:repair

# Stato generale dell'installazione
docker exec -u www-data CONTAINER php occ status
```

File (ripristino da snapshot / backup)

Quando si ripristinano file direttamente sul filesystem, Nextcloud non li "vede" finché non si rilancia la scansione che risincronizza il database con il disco.

```
docker exec -u www-data CONTAINER php occ files:scan --all
docker exec -u www-data CONTAINER php occ files:scan UTENTE
docker exec -u www-data CONTAINER php occ files:scan --path="/UTENTE/files/Cartella"
docker exec -u www-data CONTAINER php occ files:cleanup # rimuove voci orfane dal DB
```

“ **Nota:** il `--path` parte sempre da `/utente/files/...`, non dal percorso assoluto sul disco. I file ripristinati devono trovarsi nel volume dati montato nel container (di solito `/var/www/html/data`) e appartenere a `www-data`.

Correggere il proprietario dei file ripristinati

```
docker exec -u www-data CONTAINER find /var/www/html/data/UTENTE/files/Cartella -exec chown www-data:www-data {} \;
```

Utenti

```
docker exec -u www-data CONTAINER php occ user:list
docker exec -u www-data CONTAINER php occ user:add NOMEUTENTE
docker exec -u www-data CONTAINER php occ user:resetpassword NOMEUTENTE
docker exec -u www-data CONTAINER php occ user:setting NOMEUTENTE files quota 10GB
```

App

```
docker exec -u www-data CONTAINER php occ app:list
docker exec -u www-data CONTAINER php occ app:enable NOMEAPP
docker exec -u www-data CONTAINER php occ app:disable NOMEAPP
docker exec -u www-data CONTAINER php occ app:update --all
```

Configurazione e indici database

```
# Leggi/scrivi parametri di config.php
docker exec -u www-data CONTAINER php occ config:list system
docker exec -u www-data CONTAINER php occ config:system:set key --value="valore"

# Indici e colonne mancanti (warning tipici nella pagina "Panoramica")
docker exec -u www-data CONTAINER php occ db:add-missing-indices
docker exec -u www-data CONTAINER php occ db:add-missing-columns
docker exec -u www-data CONTAINER php occ db:convert-filecache-bigint
```

Cron / anteprime

```
docker exec -u www-data CONTAINER php occ background:cron # imposta cron come metodo
job
docker exec -u www-data CONTAINER php occ preview:generate-all # rigenera anteprime
```

Suggerimento: alias per scrivere meno

Crea un alias nella shell dell'host per non riscrivere ogni volta la parte lunga:

```
alias nocc='docker exec -u www-data CONTAINER php occ'
```

Poi basta scrivere, ad esempio:

```
nocc files:scan --all
nocc status
```

Se usi **docker compose**, l'equivalente del comando base è:

```
docker compose exec -u www-data app php occ <comando>
```

(sostituisci `app` con il nome del servizio Nextcloud nel tuo `docker-compose.yml`).

Revision #1

Created 2026-06-03 19:37:03 UTC by Pe

Updated 2026-06-03 19:38:06 UTC by Pe