

Test Deploy Daemon



- [Status Checks](#)
 - [Register](#)
 - [Image Pull](#)
 - [Container Started](#)
 - [Heartbeat](#)
 - [Init](#)
 - [Enabled](#)
 - [Download Logs](#)
-

You can test each Daemon configuration deployment from the AppAPI Admin settings.

Deploy Daemons ?

Deploy Daemon (DaemonConfig) is an ExApps orchestration daemon.

docker_socket_proxy - Docker Socket Pr...

docker-install

Default

...

manual_install - Manual install

manual-install

docker_socket_proxy_amd - Docker Sock...

docker-install

docker_socket_proxy_cpu - Docker Socke...

docker-install

✓ Default

🧪 Test deploy

🗑 Delete

1

...

+ Register Daemon

Status Checks

The Deploy test installs a [test-deploy](#) ExApp to verify each step of the deployment process, including a hardware support check - for each compute device, there is a separate Docker image.

“ ”

The Test Deploy ExApp container is not removed after the test as it's needed for logs and status checks. You can remove it after testing from the Apps page. The Docker images are also not removed from the Daemon; you can clean up unused images with the `docker image prune` command.

Test deploy - Docker Socket Proxy CUDA



AppAPI will try to install small skeleton ExApp to verify Daemon configured correctly and deployment steps are passing.

The following Deploy test checks must be passed to succeed: (3 / 6)

✓ **Image pull (100%)**
Check if the image is successfully pulled

✓ **Container started**
Check if the image successfully pulled and container is created and started

❗ **Heartbeat**
Check for the heartbeat is finished and healthy

Container healthcheck failed

[More info](#)

i **Init step**
Wait for initialization step to finish

↓ Download ExApp logs

Start Deploy test

Register

The Register step is the first step; it checks if the ExApp is registered in Nextcloud.

Image Pull

The Image Pull step downloads the ExApp Docker image.

Possible errors:

- Image not found (e.g. not public, no image found for your hardware architecture)
- Image pull failed (e.g., due to network issues)
- Image pull timeout
- Your Docker Socket Proxy/HaRP is not configured correctly and blocks access to this Docker Engine API

See `journalctl -f -u docker.service` for more details in systemd based systems.

Container Started

The Container Started step verifies that the ExApp container is created and started successfully.

Possible errors:

- Container failed to start with GPU support (may be missing or misconfigured)
 - For NVIDIA, refer to the [NVIDIA Docker configuration docs](#).
 - For AMD, refer to the [ROCm Docker configuration docs](#).
- The ExApp issue during startup (e.g. not enough memory). The app would show repeated starting attempts in the logs.

See `docker logs nc_app_<app_id>` for more details.

Heartbeat

The Heartbeat step checks if the container's health check is finished and the container is healthy. The ExApp might have additional pre-configuration logic during this step.

Possible errors:

- ExApp failed to start a web server, e.g., if the port is already in use (this should be visible in the container logs)
- ExApp heartbeat_count keeps increasing, this may indicate that the ExApp couldn't start properly
- Nextcloud can not reach the ExApp container, e.g.,

- due to a network issue or a firewall (this should be visible in the server logs or the firewall logs)
- due to a "http" protocol deploy daemon. In this case, the ExApp's container listens on localhost (127.0.0.1 or ::1) which might not be reachable from the Nextcloud server and you might want to listen on a different IP address. See `OVERRIDE_APP_HOST` in [Additional options](#) in the Deploy Daemon form. This issue can be identified using this command: `lsof -i -P -n | grep LISTEN`
- For HaRP, the main Nextcloud proxy might not be configured to redirect requests to the HaRP container correctly. See the [Configuring Your Reverse Proxy](#) section in the HaRP readme.

Init

The Init step checks if the ExApp is initialized and ready to use. During the init step, the ExApp may perform downloads of extra stuff required for it.

Possible errors:

- Initialization failed (e.g., due to network issues or timeout)
- ExApp not being able to reach the Nextcloud server (e.g., due to a network issue or a firewall)

Enabled

The Enabled step checks if the ExApp is enabled and ready to use. During this step, the ExApp registers all the required and available APIs of the Nextcloud AppFramework.

Possible errors:

- ExApp did not respond to the enable request
- ExApp failed to enable due to a failure in registering AppAPI Nextcloud AppFramework APIs (this should be visible both in the container logs and in the Nextcloud logs if there are any errors)

Download Logs

You can download the logs of the last test deploy attempt container.



Downloading Docker container logs is only possible for containers using the json-file or journald logging drivers.

📦 #3

📦 cocobook-sync 📦 2026-10-28 16:28:38 UTC

📦 cocobook-sync 📦 2026-10-29 01:09:21 UTC