

# Nextcloud: Nube privada autohospedada con imagen oficial, MariaDB y Redis

---

## Introducción

Nextcloud se utiliza como plataforma principal de almacenamiento, sincronización y colaboración dentro del stack. El servicio permite centralizar archivos, compartir contenido y mantener una nube privada controlada desde la infraestructura propia.

Este despliegue utiliza la imagen oficial de Nextcloud en su variante Apache, separando la aplicación, la base de datos y la caché en contenedores independientes. La instalación se integra con el resto del entorno mediante reverse proxy, volúmenes persistentes y una base de datos MariaDB dedicada.

A diferencia de despliegues basados en imágenes de terceros, esta variante mantiene una estructura más cercana al proyecto oficial de Nextcloud. Esto simplifica la relación con la documentación upstream y reduce diferencias de rutas, comportamiento interno y gestión del contenedor.

---

## Enfoque general / Arquitectura

El despliegue queda dividido en varios componentes:

- `nextcloud`, como aplicación web principal.
- `cron`, como contenedor dedicado para ejecutar las tareas programadas de Nextcloud.
- `db`, como base de datos MariaDB.
- `redis`, como caché transaccional y sistema auxiliar de bloqueo.
- Volúmenes persistentes para aplicación, datos, base de datos y Redis.
- Red Docker dedicada para aislar el servicio.

El tráfico HTTP entra por el puerto interno publicado por Docker y queda pensado para ser consumido por un reverse proxy externo. El TLS no se gestiona dentro de Nextcloud, sino en el

proxy frontal.

La base de datos forma parte del mismo despliegue Docker, pero se mantiene separada del contenedor de aplicación. Esto permite aislar responsabilidades, facilitar copias de seguridad y evitar mezclar estado crítico dentro del contenedor web.

---

## Requisitos previos

- Docker y Docker Compose operativos.
  - Dominio o subdominio apuntando al reverse proxy.
  - Almacenamiento persistente suficiente para los datos de usuario.
  - Variables sensibles definidas en un archivo `.env`.
  - Reverse proxy configurado para publicar Nextcloud mediante HTTPS.
- 

## Desarrollo

### Qué se hizo y por qué

#### Uso de la imagen oficial de Nextcloud

Se utiliza la imagen oficial `nextcloud:${NEXT_VERSION}` en variante Apache. Esta decisión evita depender de rutas, capas de abstracción o mecanismos propios de imágenes mantenidas por terceros.

La versión se fija mediante la variable `NEXT_VERSION` para evitar actualizaciones implícitas no controladas. Esto permite decidir cuándo subir de versión y revisar previamente los cambios de Nextcloud.

#### Contenedor cron separado

Nextcloud necesita ejecutar tareas periódicas para mantenimiento interno, limpieza, procesos diferidos, trabajos de aplicaciones y operaciones programadas.

En lugar de depender de AJAX o Webcron, se define un contenedor `cron` usando la misma imagen que la aplicación principal y el entrypoint `/cron.sh`. Ambos contenedores comparten los mismos volúmenes, por lo que el contenedor de cron opera sobre la misma instalación de Nextcloud.

#### MariaDB como base de datos dedicada

MariaDB se despliega como contenedor independiente dentro del mismo stack. La aplicación accede a la base de datos mediante el hostname interno `db`, resuelto dentro de la red Docker del despliegue.

Separar la base de datos del contenedor de aplicación facilita backups, actualizaciones y mantenimiento. También evita que el estado crítico de la instancia dependa del ciclo de vida del contenedor web.

## Redis para caché y bloqueo

Redis se incluye como servicio dedicado y protegido mediante contraseña. Nextcloud lo utiliza como caché auxiliar y para mejorar el bloqueo de archivos, reduciendo problemas en operaciones concurrentes.

La persistencia de Redis se mantiene en un volumen local mediante `appendonly`, suficiente para conservar estado operativo sin acoplarlo al contenedor.

## Reverse proxy externo

El contenedor publica HTTP en el puerto `8040`, pero la URL final queda servida por HTTPS desde el reverse proxy.

Las variables `OVERWRITEHOST`, `OVERWRITEPROTOCOL` y `TRUSTED_PROXIES` permiten que Nextcloud genere URLs correctas, detecte el esquema HTTPS real y confíe en las cabeceras enviadas por el proxy.

## Volúmenes persistentes

La instalación separa el código y configuración de Nextcloud, los datos de usuario, la base de datos y Redis. Esto permite recrear contenedores sin perder estado.

También se contemplan montajes adicionales bajo `/mnt` para exponer almacenamiento externo dentro de Nextcloud mediante la aplicación de almacenamiento externo o mediante rutas locales controladas.

---

# Configuración utilizada

Los archivos de configuración se mantienen versionados en Gitea.

## [Configuración de Nextcloud en Gitea](#)

No se incrusta la configuración completa dentro del artículo para evitar duplicidades, configuraciones obsoletas y errores de copia.

---

# Ajustes posteriores en `config.php`

Tras la instalación inicial se añaden varios parámetros de localización y mantenimiento para adaptar la instancia al entorno:

- Idioma por defecto en español.

- Locale del sistema en `es_ES.UTF-8`.
- Zona horaria `Europe/Madrid`.
- Región telefónica por defecto `ES`.
- Ventana de mantenimiento programada durante la madrugada.
- Modo mantenimiento desactivado explícitamente tras completar la instalación.

El bloque añadido en `config.php` queda así:

```
'maintenance_window_start' => 2,  
'default_phone_region' => 'ES',  
'maintenance' => false,  
'default_language' => 'es',  
'default_locale' => 'es_ES.UTF-8',  
'force_locale' => 'es_ES.UTF-8',  
'default_timezone' => 'Europe/Madrid',  
'force_language' => 'es',
```

Estos ajustes evitan avisos posteriores en el panel de administración y hacen que la instancia quede alineada con el uso real del servicio.

---

## Validación

Comprobaciones mínimas tras el despliegue:

- Contenedores `nextcloud`, `cron`, `db` y `redis` en estado `running`.
  - Acceso correcto a la interfaz web desde el dominio configurado.
  - Nextcloud detectando correctamente HTTPS detrás del reverse proxy.
  - Conexión funcional entre Nextcloud y MariaDB.
  - Conexión funcional entre Nextcloud y Redis.
  - Tareas de cron ejecutándose sin avisos en el panel de administración.
  - Persistencia de datos tras reiniciar o recrear contenedores.
  - Logs sin errores recurrentes de base de datos, Redis, permisos o proxy.
- 

## Decisiones importantes o problemas detectados

**La versión de Nextcloud queda fijada**

No se utiliza `latest`. La versión se define en `.env` mediante `NEXT_VERSION`, permitiendo controlar actualizaciones y evitar saltos inesperados.

### **La base de datos no debe recrearse sin backup**

El volumen de MariaDB contiene estado crítico. Cualquier cambio destructivo sobre ese volumen implica pérdida de la instancia salvo que exista backup válido.

### **Cron no es opcional en una instancia estable**

Para una instalación mantenida en el tiempo, el contenedor de cron evita depender de ejecuciones disparadas por visitas web. Esto mejora el comportamiento de trabajos internos y reduce avisos administrativos.

### **Redis debe compartir configuración con aplicación y cron**

El contenedor `cron` necesita las mismas variables de Redis y base de datos que el contenedor principal. Si no las recibe, algunas tareas pueden ejecutarse con configuración incompleta o generar errores difíciles de relacionar.

### **Los montajes externos deben revisarse con permisos coherentes**

Los directorios montados bajo `/mnt` quedan disponibles dentro del contenedor, pero deben tener permisos compatibles con el usuario interno del servidor web. Si los permisos no son correctos, Nextcloud podrá ver rutas pero no operar correctamente sobre ellas.

---

## Resumen breve

Nextcloud queda desplegado con la imagen oficial Apache, MariaDB dedicada, Redis y un contenedor separado para cron.

La arquitectura separa aplicación, base de datos, caché y datos persistentes. El TLS se delega al reverse proxy y la versión de Nextcloud se fija desde `.env` para evitar actualizaciones no controladas.

El despliegue prioriza mantenimiento sencillo, compatibilidad con la documentación oficial y una estructura clara para backups, actualizaciones y diagnóstico.

---

## Referencias

- [Documentación oficial de Nextcloud](#)
- [Imagen oficial de Nextcloud en Docker Hub](#)

- [Repositorio oficial de la imagen Docker de Nextcloud](#)
- 

Revision #1

Created 2026-06-09 12:34:14 UTC by Juan Francisco

Updated 2026-06-09 12:53:43 UTC by Juan Francisco