Nextcloud + Traefik

Requerimientos previos

- Docker instalado
- Instancia de Traefik instalada y corriendo
- Red de Docker creada para conectar contenedor con Traefik
- Opcionalmente, usa un middleware configurado en Traefik:
  - autoDetectContentType@file

Pasos

1. Preparar dominio/subdominio de la instancia a configurar
2. Crear archivos de configuración
   1. Archivo de variables de entorno: `.env`
   2. Archivo para Docker Compose: `compose.yaml`
   3. Archivo de configuración personalizada de MariaDB/MySQL: `my.cnf`
   4. Archivo de configuración personalizada de PHP: `php.ini`
3. Cambiar valores de acuerdo a las necesidades de la instancia
4. Levantar contenedor (`docker composer up`)
5. Visitar aplicación y comprobar que esté funcionando correctamente

Configuración previa

> Por documentar

Archivos de configuración

.env

```
INSTANCE_URL=ejemplo.com
APP_CONTAINER_NAME=nextcloud-app
APP_CONTAINER_USER=www-data
MYSQL_ROOT_PASSWORD=abc123456
```

```
REDIS_HOST_PASSWORD=abc123456
MYSQL_DATABASE=nextcloud_db
MYSQL_USER=nextcloud_db_admin
MYSQL_PASSWORD=abc123456
VOLUME_PATH=./data
```

compose.yaml

> En la configuración de `entrypoints` y `certresolver` cambiar de acuerdo a los valores configurados en Traefik.

```
services:
  nextcloud-db:
      image: mariadb:lts
      container_name: nextcloud-db
      command: --transaction-isolation=READ-COMMITTED --log-bin=ROW --skip-innodb-read-only-
compressed
      restart: unless-stopped
      volumes:
          - ${VOLUME_PATH}/db:/var/lib/mysql
          - ./my.cnf:/etc/mysql/my.cnf
      environment:
          - MYSQL_ROOT_PASSWORD=${MYSQL_ROOT_PASSWORD}
          - MYSQL_DATABASE=${MYSQL_DATABASE}
          - MYSQL_USER=${MYSQL_USER}
          - MYSQL_PASSWORD=${MYSQL_PASSWORD}
      networks:
          - traefik
  nextcloud-redis:
      image: redis:alpine
      container_name: nextcloud-redis
      hostname: nextcloud-redis
      restart: unless-stopped
      command: redis-server --requirepass ${REDIS_HOST_PASSWORD}
      networks:
          - traefik
  nextcloud-app:
      image: nextcloud
      container_name: nextcloud-app
```

```
      restart: unless-stopped
      depends_on:
          - nextcloud-db
          - nextcloud-redis
      environment:
          - REDIS_HOST=nextcloud-redis
          - REDIS_HOST_PASSWORD=${REDIS_HOST_PASSWORD}
          - MYSQL_HOST=nextcloud-db
          - MYSQL_DATABASE=${MYSQL_DATABASE}
          - MYSQL_USER=${MYSQL_USER}
          - MYSQL_PASSWORD=${MYSQL_PASSWORD}
          - OVERWRITEPROTOCOL=https
          - OVERWRITEHOST=${INSTANCE_URL}
          - TRUSTED_PROXIES=127.0.0.1
      volumes:
          - ${VOLUME_PATH}/files:/var/www/html
          - ./nextcloud.ini:/usr/local/etc/php/conf.d/nextcloud.ini
      labels:
          - "traefik.enable=true"
          - "traefik.http.middlewares.nc-rep1.redirectregex.permanent=true"
          - "traefik.http.middlewares.nc-rep1.redirectregex.regex=https://(.*)/.well-
known/(?:card|cal)dav"
          - "traefik.http.middlewares.nc-
rep1.redirectregex.replacement=https://$${1}/remote.php/dav"
          - "traefik.http.routers.nextcloud-secure.rule=Host(`${INSTANCE_URL}`)"
          - "traefik.http.routers.nextcloud-secure.entrypoints=https"
          - "traefik.http.routers.nextcloud-secure.middlewares=nc-rep1@docker,secHeaders@file"
          - "traefik.http.routers.nextcloud-secure.tls=true"
          - "traefik.http.routers.nextcloud-secure.tls.certresolver=tls"
          - "traefik.http.services.nextcloud-secure.loadbalancer.server.port=80"
          - "traefik.http.services.nextcloud-secure.loadbalancer.passHostHeader=true"
      networks:
          - traefik
  nextcloud-cron:
      container_name: nextcloud-cron
      image: nextcloud
      restart: unless-stopped
      volumes:
          - ${VOLUME_PATH}/files:/var/www/html
      entrypoint: /cron.sh
```

```
      networks:
          - traefik
networks:
  traefik:
    external: true
```

my.cnf

> Configuración para reducir el consumo de memoria de la base de datos. Es opcional. Se debe ajustar a las necesidades de la instancia.

```
# MariaDB configuración personalizada
# para instancia pequeña, de uso personal
# y principalmente almacenamiento de archivos

[mysqld]
bulk_insert_buffer_size = 0
connect_timeout         = 10
innodb_buffer_pool_size = 75M
innodb_log_buffer_size  = 1M
max_allowed_packet      = 25M
max_connections         = 100
max_heap_table_size     = 16K
query_cache_limit       = 128K
query_cache_size        = 512K
sort_buffer_size        = 10M
thread_cache_size       = 4M
tmp_table_size          = 1K
wait_timeout            = 600
```

nextcloud.ini

> Configuración para ampliar la capacidad de subida de archivos. Es opcional. Se debe ajustar a las necesidades de la instancia.

```
memory_limit=1G
upload_max_filesize=1G
post_max_size=1G
```

```
max_input_time=3600
max_execution_time=3600
```

Scripts útiles

> Colocar en la misma carpeta que el archivo compose.yaml y el archivo .env

occ.sh

```
#!/bin/bash
#
# Ejemplo de uso:
# ./occ.sh app:update --all
#
source .env
sudo docker exec -ti --user $APP_CONTAINER_USER $APP_CONTAINER_NAME /var/www/html/occ $@
```

fix-permission.sh

```
#!/bin/bash
source .env
sudo docker exec -u root $APP_CONTAINER_NAME sh -c 'chmod 755 /var/www/html/config'
sudo docker exec -u root $APP_CONTAINER_NAME sh -c 'chown -R www-data:www-data
/var/www/html/config'
sudo docker exec -u root $APP_CONTAINER_NAME sh -c 'ls -lsa /var/www/html'
```

Revisión #6
Creado 2025-08-26 01:29:15 UTC por IM
Actualizado 2026-03-07 17:24:32 UTC por IM