Nextcloud

Cette documentation détaille l'installation d'une instance Nextcloud All-In-One (AIO) optimisée. La particularité de cette installation réside dans l'utilisation d'un stockage déporté, d'un reverse-proxy Caddy et de certificats SSL locaux via mkcert.

Prérequis système

Avant de commencer, nous installons les outils de base nécessaires à la gestion des dépôts, du réseau et des conteneurs.

```
apt install curl git wget rsync docker-compose-plugin
```

Configuration du stockage dédié

Pour garantir les performances, nous dédions un disque entier (`/dev/sdb`) aux données.

```
fdisk -l
mkfs -t ext4 /dev/sdb
sudo mkdir -p /mnt/sdb
lsblk -no UUID /dev/sdb  # Note l'UUID
echo "UUID=xxx /mnt/sdb ext4 defaults 0 2" >> /etc/fstab
sudo mount -a
```

Docker (sur `/mnt/sdb`)

```
# Installation Docker
curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o
/etc/apt/keyrings/docker.gpg
echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg]
https://download.docker.com/linux/debian $(lsb_release -cs) stable" | sudo tee
/etc/apt/sources.list.d/docker.list > /dev/null
sudo apt update && sudo apt install docker-ce docker-ce-cli docker-compose-plugin
```

```
containerd.io

# Docker sur /mnt/sdb
sudo systemctl stop docker docker.socket
sudo mkdir -p /mnt/sdb/docker

sudo tee /etc/docker/daemon.json <<EOF
{
  "data-root": "/mnt/sdb/docker"
}
EOF

sudo rsync -aP /var/lib/docker/ /mnt/sdb/docker/
sudo systemctl start docker docker.socket
```

mkcert SSL

```
# Installation mkcert
wget $(curl -s https://api.github.com/repos/FiloSottile/mkcert/releases/latest | grep
browser_download_url | grep linux-amd64 | cut -d '"' -f 4)
sudo mv mkcert* /usr/local/bin/mkcert && sudo chmod +x /usr/local/bin/mkcert
mkcert -install
mkcert cloud.eyrode.lan

# Permissions
sudo mkdir -p /var/opt/mkcert
sudo mv cloud.eyrode.lan*.pem /var/opt/mkcert/
sudo chown -R 1000:1000 /var/opt/mkcert
sudo chmod 644 /var/opt/mkcert/*.pem
sudo chmod 600 /var/opt/mkcert/*-key.pem
```

Nextcloud AIO (HTTPS + Caddy)

```
sudo mkdir -p /var/opt/nextcloud/all-in-one
cd /var/opt/nextcloud/all-in-one
git clone https://github.com/nextcloud/all-in-one.git .
```

`compose.yaml` **FINAL** (copie-colle complet) :

```
name: nextcloud-aio
services:
  nextcloud-aio-mastercontainer:
    image: ghcr.io/nextcloud-releases/all-in-one:latest
    init: true
    restart: always
    container_name: nextcloud-aio-mastercontainer
    volumes:
      - nextcloud_aio_mastercontainer:/mnt/docker-aio-config
      - /var/run/docker.sock:/var/run/docker.sock:ro
      - /var/opt/mkcert:/mnt/certs:ro
    ports:
      - 8080:8080
    environment:
      APACHE_PORT: 11000
      APACHE_IP_BINDING: 0.0.0.0
      NEXTCLOUD_DOMAIN: cloud.eyrode.lan
      SKIP_DOMAIN_VALIDATION: true
      SSL_CERTIFICATE_FILE: /mnt/certs/cloud.eyrode.lan.pem
      SSL_CERTIFICATE_KEY_FILE: /mnt/certs/cloud.eyrode.lan-key.pem

  caddy:
    image: caddy:alpine
    restart: always
    container_name: caddy
    volumes:
      - caddy_certs:/certs
      - caddy_config:/config
      - caddy_data:/data
      - caddy_sites:/srv
      - /var/opt/mkcert:/var/opt/mkcert:ro
    network_mode: host
    configs:
      - source: Caddyfile
        target: /etc/caddy/Caddyfile

configs:
  Caddyfile:
```

```
    content: |
      cloud.eyrode.lan {
        reverse_proxy localhost:11000
        tls /var/opt/mkcert/cloud.eyrode.lan.pem /var/opt/mkcert/cloud.eyrode.lan-key.pem
      }

volumes:
  nextcloud_aio_mastercontainer:
    name: nextcloud_aio_mastercontainer
  caddy_certs:
  caddy_config:
  caddy_data:
  caddy_sites:
```

Lancement :

```
docker compose up -d
```

Dépannage (Si nécessaire)

Port 443 bloqué :

```
sudo systemctl restart docker
docker system prune -af
docker compose up -d
```

Interface AIO : `https://cloud.eyrode.lan:8080` **Clique "Start all containers" → 5min** → tout vert

Accès Final

```
https://cloud.eyrode.lan           # Nextcloud (cadenas vert mkcert)
https://cloud.eyrode.lan:8080      # Interface AIO
```

Logs de diagnostic

```
docker compose logs nextcloud-aio-mastercontainer
docker compose logs caddy
docker ps  # Tous les conteneurs doivent être "Up"
```

Récupération Mot de Passe Admin

Méthode 1 : Interface AIO (RECOMMANDÉE) https://cloud.eyrode.lan:8080 → "Access details" → Password admin affiché !

Méthode 2 : Reset CLI (syntaxe correcte)

```
docker exec -u www-data -e OC_PASS="MonSuperPass123!" nextcloud-aio-nextcloud php occ user:add
tanny --password-from-env --group=admin --display-name="Eythan"
```

Revision #1
Created 2026-09-23 09:05:41 UTC by Eythan
Updated 2026-09-23 09:05:41 UTC by Eythan