

TP Docker

Nom et Prénom : AGBIDIHO Mario

Le TP consiste à mettre en place une solution NextCloud

Etape 1 - déployer un NextCloud simple

“La première étape consiste à déployer un conteneur Nextcloud lié à une base de données Postgres ou Maria DB. Le conteneur Nextcloud doit être connecté à la base de données avec les bonnes informations d'identification (user, password, host, port, etc.). Les conteneurs peuvent être lancés manuellement. A ce stade les données peuvent ne pas être pérennisées à l'aide de l'utilisation de volumes (que ce soit pour la base ou le drive Nextcloud).”

1.1 Créer un réseau Docker

```
PS C:\Users\mario> docker network create nextcloud-net
1907b2579ccafa6efc8e2a9f2c0f3b889997ffbee9edc48d4f1a0142b2b2e78e
PS C:\Users\mario>
```

- J'ai commencé par créer un réseau interne Docker nommé nextcloud-net pour que NextCloud et PostgreSQL puissent communiquer entre eux par leurs noms de conteneurs.

1.2 Lancer PostgreSQL

```
PS C:\Users\mario> docker run -d --name nextcloud-db --network nextcloud-net -e POSTGRES_DB=nextcloud -e POSTGRES_USER=nextclouduser -e POSTGRES_P
ASSWORD=securepassword postgres:alpine
Unable to find image 'postgres:alpine' locally
alpine: Pulling from library/postgres
098fcdec177a: Pull complete
f08522f70cd2: Pull complete
2c4f615286e4: Pull complete
8445e5ffc9d7: Pull complete
fc07684b16b8: Pull complete
1a9f81ee9ac0: Pull complete
01e5a9a905b9: Pull complete
692bc6b67a72: Pull complete
928879f614f6: Pull complete
e8dbbe71bef5: Pull complete
Digest: sha256:bcb90dc18910057ff49ce2ea157d8a0d534964090d39af959df41083f18c3318
Status: Downloaded newer image for postgres:alpine
5fb97e830551539295bf73762c68edfb79f14ea50a509f60d98609c44a22877
PS C:\Users\mario>
```

- Lance un conteneur PostgreSQL nommé nextcloud-db avec :

- (POSTGRES_DB=nextcloud) qui est le nom de la base
- (POSTGRES_USER=nextclouduser) le nom de l'utilisateur
- (POSTGRES_PASSWORD=securepassword) le mot de passe

On constate également que la commande PostgreSQL s'est exécutée correctement et aussi que l'image postgres : alpine a bien été téléchargée et lancée.

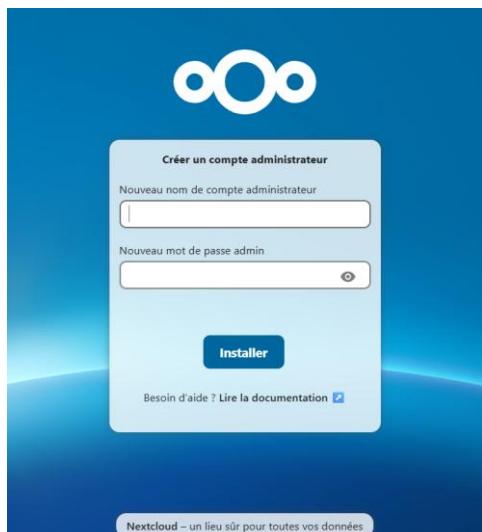
1.3 Lancer NextCloud

```
PS C:\Users\mario> docker run -d --name nextcloud-app --network nextcloud-net -p 8080:80 -e POSTGRES_DB=nextcloud -e POSTGRES_USER=nextclouduser -e POSTGRES_PASSWORD=securepassword -e POSTGRES_HOST=nextcloud-db nextcloud
Unable to find image 'nextcloud:latest' locally
latest: Pulling from library/nextcloud
61b04524b9bc: Pulling fs layer
b8a362458600: Pulling fs layer
336e5048d4f7: Pulling fs layer
9720375f9774: Pull complete
c62dbdac9dec: Pull complete
bd5886df122f: Pull complete
682ee37b9da4: Pull complete
d54a66f47a09: Pull complete
61320b01ae5e: Pull complete
Digest: sha256:e775d46d9888d87d11c76bd8c5d19561e99e6975052c4c5bd7f2610392b0cbc7
Status: Downloaded newer image for nextcloud:latest
01208473c4b3b97b1a0d29b4c25ace0c60dde70ade44b0617cdaf74f4f5eef4b
PS C:\Users\mario>
```

➤ Cette commande :

- Lance un conteneur NextCloud (nextcloud-app)
 - Le relie au même réseau que la base (nextcloud-net)
 - Expose le site sur <http://localhost:8080>
 - Configure la connexion à la base Postgres (via les variables d'environnement)
- On remarque aussi que l'image Docker de **NextCloud** s'est bien téléchargée et que mon conteneur a été lancé avec succès

Pour une petite vérification j'ai ouvert mon navigateur à <http://localhost:8080> et voilà le résultat :



. On voir bien que NextCloud fonctionne bien car je suis arrivé à l'écran d'installation

Etape 2 - sécuriser ses données avec les volumes

“Sécuriser ses données avec les volumes La deuxième étape consiste à pérenniser les données de la base et du cloud à l'aide de volumes. Vous devrez alors montrer qu'en supprimant ses conteneurs et en les recréant, les données sont bien présentes”

2.1 Supprimer les anciens conteneurs

```
PS C:\Users\mario> docker rm -f nextcloud-app nextcloud-db  
nextcloud-app  
nextcloud-db  
PS C:\Users\mario>
```

Les conteneurs NextCloud et PostgreSQL ont bien été supprimés

2.2 Créer deux volumes docker

```
PS C:\Users\mario> docker volume create nextcloud-data  
nextcloud-data  
PS C:\Users\mario> docker volume create postgres-data  
postgres-data  
PS C:\Users\mario>
```

Ces volumes vont stocker les fichiers NextCloud et les données de la base PostgreSQL.

2.3 Recréer PostgreSQL avec un volume

```
PS C:\Users\mario> docker run -d --name nextcloud-db --network nextcloud-net -e POSTGRES_DB=nextcloud -e POSTGRES_USER=nextclouduser -e POSTGRES_PASSWORD=securepassword -v postgres-data:/var/lib/postgresql/data postgres:alpine
7ab6bdc19b411cfaaadd464cfb8cc6bbf13198227615fc8d3cebc9607100961
PS C:\Users\mario>
```

(-v postgres-data:/var/lib/postgresql/data)
monte le volume postgres-data dans le dossier où PostgreSQL stocke ses données.

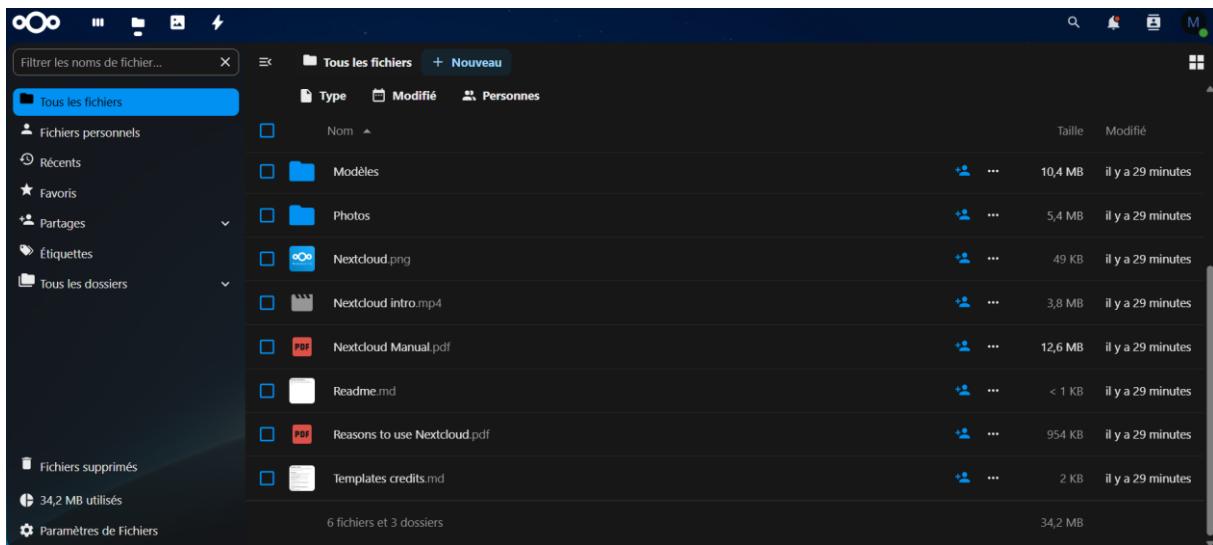
2.4 Recréer NextCloud avec un volume

```
PS C:\Users\mario> docker run -d --name nextcloud-app --network nextcloud-net -p 8080:80 -e POSTGRES_DB=nextcloud -e POSTGRES_USER=nextclouduser -e POSTGRES_PASSWORD=securepassword -e POSTGRES_HOST=nextcloud-db -v nextcloud-data:/var/www/html nextcloud
c2d07f2d0489e0b9cd6ccde342780707676cbf5f3192d59d88bcd9893279970
PS C:\Users\mario> |
```

(-v nextcloud-data:/var/www/html)
monte le volume nextcloud-data dans le répertoire de l'application NextCloud.

➤ Vérification :

J'accède à NextCloud (<http://localhost:8080>)
J'ai créé un compte admin et ensuite je me suis connecté



- J'ajoute un fichier test

Documents	Folder			1,1 MB	il y a 44 minutes
Modèles	Folder			10,4 MB	il y a 44 minutes
Photos	Folder			5,4 MB	il y a 44 minutes
Fichier_test.txt	Text			0 KB	il y a 16 minutes
Nextcloud.png	Image			49 KB	il y a 44 minutes
Nextcloud intro.mp4	Video			3,8 MB	il y a 44 minutes
Nextcloud Manual.pdf	PDF			12,6 MB	il y a 44 minutes
Readme.md	Text			< 1 KB	il y a 44 minutes
Reasons to use Nextcloud.pdf	PDF			954 KB	il y a 44 minutes

- Je supprime les conteneurs :

```
PS C:\Users\mario> docker rm -f nextcloud-app nextcloud-db
nextcloud-app
nextcloud-db
PS C:\Users\mario> []
```

- Je recrée les deux conteneurs avec les mêmes volumes

```
PS C:\Users\mario> docker run -d --name nextcloud-db --network nextcloud-net -e POSTGRES_DB=nextcloud -e POSTGRES_USER=nextclouduser -e POSTGRES_PASSWORD=securepassword -v postgres-data:/var/lib/postgresql/data postgres:alpine
e8523fd396b54e5d5c414c0c6c55ab35d1e74c9ac22ea88c78dc958c84ff8c03
PS C:\Users\mario> docker run -d --name nextcloud-app --network nextcloud-net -p 8080:80 -e POSTGRES_DB=nextcloud -e POSTGRES_USER=nextclouduser -e POSTGRES_PASSWORD=securepassword -e POSTGRES_HOST=nextcloud-db -v nextcloud-data:/var/www/html nextcloud
a8e2c1ba55f250bdd70a5bae209d94ef1195e6f6be5c6f735afc1081447a3e77
PS C:\Users\mario> []
```

- Je me reconnecte-toi à <http://localhost:8080>

Type	Nom	Taille	Modifié
Documents	Documents	1,1 MB	il y a 1 heure
Modèles	Modèles	10,4 MB	il y a 1 heure
Photos	Photos	5,4 MB	il y a 1 heure
Texte	Fichier_test.txt	0 KB	il y a 33 minutes
Image	Nextcloud.png	49 KB	il y a 1 heure
Video	Nextcloud intro.mp4	3,8 MB	il y a 1 heure
PDF	Nextcloud Manual.pdf	12,6 MB	il y a 1 heure

Je n'ai pas eu besoins de reconfigurer NextCloud et mes fichiers, compte, etc. sont toujours là

Etape 3 - structurer un déploiement avec Compose

“La troisième étape consiste à déployer cela dans un fichier docker compose. Vous pouvez utiliser docker compose ou Swarm, au choix : indiquez simplement avec quel outil il faut déployer votre projet. Le conteneur de la base de données ne doit pas exposer son port vers l'extérieur du stack.”

3.1 Je commence à créer un dossier :  nextcloud-compose

Ensuite dans ce même dossier Je créé un fichier docker-compose.yml qui : Déploie NextCloud et PostgreSQL

```
C:\ > Users > mario > Downloads > docker-compose.yml > ...
      docker-compose.yml - The Compose specification establishes a standard for the definition of multi-container platform-agnostic applications (comp...
1   version: '3.8'
2
3   services:
4     db:
5       image: postgres:alpine
6       container_name: nextcloud-db
7       restart: unless-stopped
8       environment:
9         POSTGRES_DB: nextcloud
10        POSTGRES_USER: nextclouduser
11        POSTGRES_PASSWORD: securepassword
12       volumes:
13         - postgres-data:/var/lib/postgresql/data
14       networks:
15         - nextcloud-net
16       # Port non exposé volontairement
17
18   app:
19     image: nextcloud
20     container_name: nextcloud-app
21     restart: unless-stopped
22     ports:
23       - "8080:80" # Expose seulement l'app NextCloud
24     environment:
25       POSTGRES_DB: nextcloud
26       POSTGRES_USER: nextclouduser
27       POSTGRES_PASSWORD: securepassword
28       POSTGRES_HOST: db
29
30   volumes:
31     - nextcloud-data:/var/www/html
32   networks:
33     - nextcloud-net
34
35   volumes:
36     postgres-data:
37     nextcloud-data:
38
39   networks:
40     - nextcloud-net:
```

```
▷ Run Service
18   app:
19     image: nextcloud
20     container_name: nextcloud-app
21     restart: unless-stopped
22     ports:
23       - "8080:80" # Expose seulement l'app NextCloud
24     environment:
25       POSTGRES_DB: nextcloud
26       POSTGRES_USER: nextclouduser
27       POSTGRES_PASSWORD: securepassword
28       POSTGRES_HOST: db
29
30   volumes:
31     - nextcloud-data:/var/www/html
32   networks:
33     - nextcloud-net
34
35   volumes:
36     postgres-data:
37     nextcloud-data:
38
39   networks:
40     - nextcloud-net:
```

➤ Mon raisonnement :

- Services : définit deux services : db (la base) et app (NextCloud)
- La base n'a **aucun port** : donc **elle n'est pas accessible directement depuis l'extérieur** (sécurité renforcée)

- Volumes : garantit la **persistence des données**
- Networks : crée un réseau interne pour que les services se voient
- Restart : unless-stopped permet un redémarrage automatique en cas de redémarrage de Docker ou du système

➤ Enfin on peut démarrer les conteneurs avec Docker Compose :

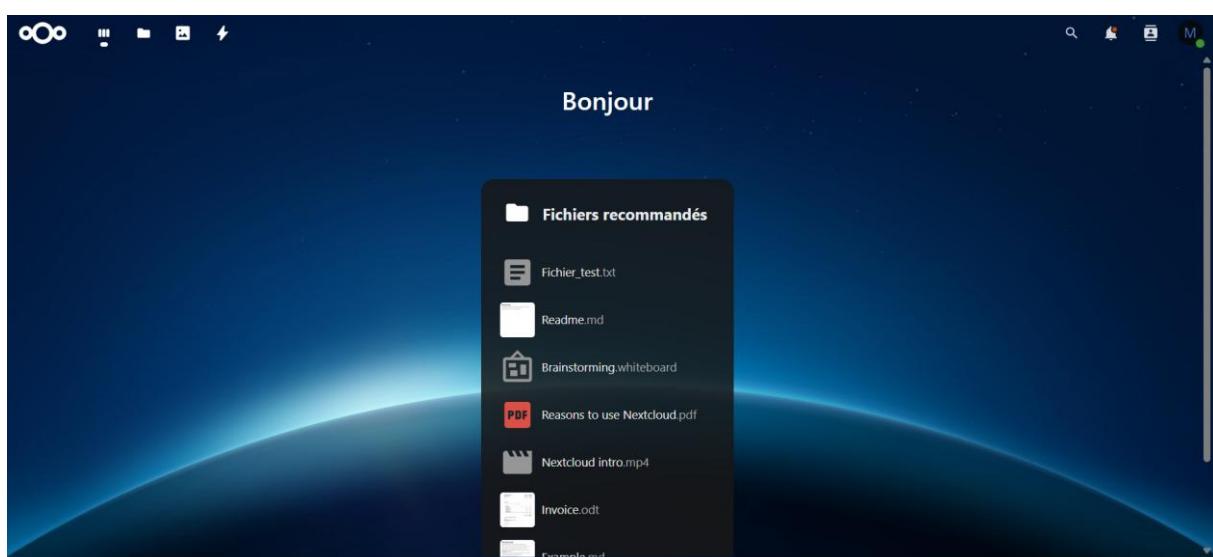
```
PS C:\Users\mario> cd "C:\Users\mario\Documents\nextcloud-compose"
PS C:\Users\mario\Documents\nextcloud-compose> docker compose up -d
time="2025-06-09T01:24:24+02:00" level=warning msg="C:\\\\Users\\\\mario\\\\Documents\\\\nextcloud-compose\\\\docker-compose.yml: the attribute `version` is
obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 3/5
  ✓ Network nextcloud-compose_nextcloud-net    Created          0.1s
  ✓ Volume "nextcloud-compose_postgres-data"   Created          0.0s
  ✓ Volume "nextcloud-compose_nextcloud-data"   Created          0.0s
- Container nextcloud-db                      Creating        0.1s
- Container nextcloud-app                     Creating        0.1s
Error response from daemon: Conflict. The container name "/nextcloud-db" is already in use by container "e8523fd396b54e5d5c414c0c6c55ab35d1e74c9ac
22ea88c78dc958c84ff8c03". You have to remove (or rename) that container to be able to reuse that name.
PS C:\Users\mario\Documents\nextcloud-compose>
```

Vérifions que tout fonctionne :

- Je commence par afficher les conteneurs actifs

```
PS C:\Users\mario\Documents\nextcloud-compose> docker ps
CONTAINER ID   IMAGE      COMMAND           CREATED          STATUS          PORTS          NAMES
a8e2c1ba55f2   nextcloud   "/entrypoint.sh apac..."   2 hours ago   Up 2 hours   0.0.0.0:8080->80/tcp   nextcloud-app
e8523fd396b5   postgres:alpine   "docker-entrypoint.s..."   2 hours ago   Up 2 hours   5432/tcp       nextcloud-db
PS C:\Users\mario\Documents\nextcloud-compose>
```

- J'ouvre ton navigateur sur : <http://localhost:8080>



Et bim on voit l'interface NextCloud prête à l'emploi

Etape 4 - explorez plus loin

“La dernière étape consiste à ajouter de petits outils complémentaires comme : adminer pour accéder à la base de données portainer pour administrer notre stack uptimekuma pour superviser notre stack un diagnostic Docker Bench Security la mise en place d'une taille maximum pour les volumes Docker la mise en place de Healthchecks Choisissez au moins deux éléments de la liste ci-dessus ou autres idées que vous pourriez avoir pour aller plus loin. Rendu attendu : explications, code éventuel, et captures d'écran illustrant le travail que vous avez réalisé”

1. Adminer – accéder à la base PostgreSQL :

Je fais une petite modification dans mon docker-compose.yml en lui ajoutant ces lignes de codes

```
adminer:
  image: adminer
  container_name: adminer
  restart: unless-stopped
  ports:
    - "8081:8080"
  networks:
    - nextcloud-net
```

2. Portainer – interface graphique pour gérer tes conteneurs

```
portainer:
  image: portainer/portainer-ce
  container_name: portainer
  restart: unless-stopped
  ports:
    - "9000:9000"
  volumes:
    - /var/run/docker.sock:/var/run/docker.sock
  networks:
    - nextcloud-net
```

Démarrage de la stack mise à jour

```
PS C:\Users\mario\Documents\nextcloud-compose> docker compose up -d
time="2025-06-09T02:06:23+02:00" level=warning msg="C:\\\\Users\\\\mario\\\\Documents\\\\nextcloud-compose\\\\docker-compose.yml: the attribute `version` is
obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 2/2
  ✓ Container nextcloud-app  Started                               0.7s
  ✓ Container nextcloud-db   Started                               0.8s
PS C:\Users\mario\Documents\nextcloud-compose>
```

Les conteneurs Nextcloud et PostgreSQL ont bien démarré

Le fichier docker-compose.yml complet se trouve sur mon GitHub :

<https://github.com/marioagbidiho/Tp-Docker>