

# Doc Job07

---

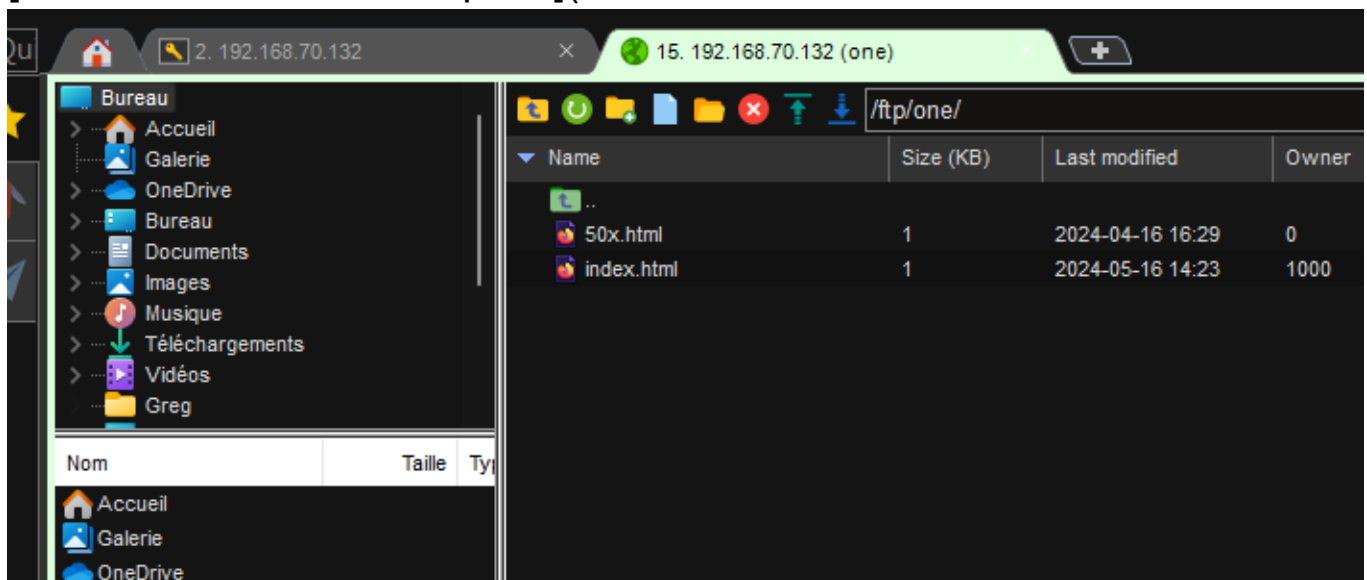
created : 24-06-02 21:30

modified : 24-06-02 21:30

tags : [note]

References :

[Générateur docker-compose](



)

[Doc docker-compose](#)

<https://zomro.com/blog/faq/317-kak-ustanovit-nginx-v-docker-compose>

Docker Compose est un outil qui permet de définir et de gérer des applications multi-conteneurs Docker. Avec Docker Compose, vous utilisez un fichier YAML pour configurer les services de votre application, puis, avec une seule commande, vous pouvez créer et démarrer tous les services à partir de cette configuration.

## Docker Compose

Conteneur



Conteneur



Je vais créer mon fichier yml de cette manière, pour combiner 2 conteneurs:

```
version: "3"

services:
  web:
    image: nginx
    volumes:
      #Créer un volume "web-data" et le monte dans le
      conteneur au chemin indiqué
      - web-data:/usr/share/nginx/html
    ports:
      - "8080:80"
    environment:
      - NGINX_HOST=foobar.com
      - NGINX_PORT=80
```

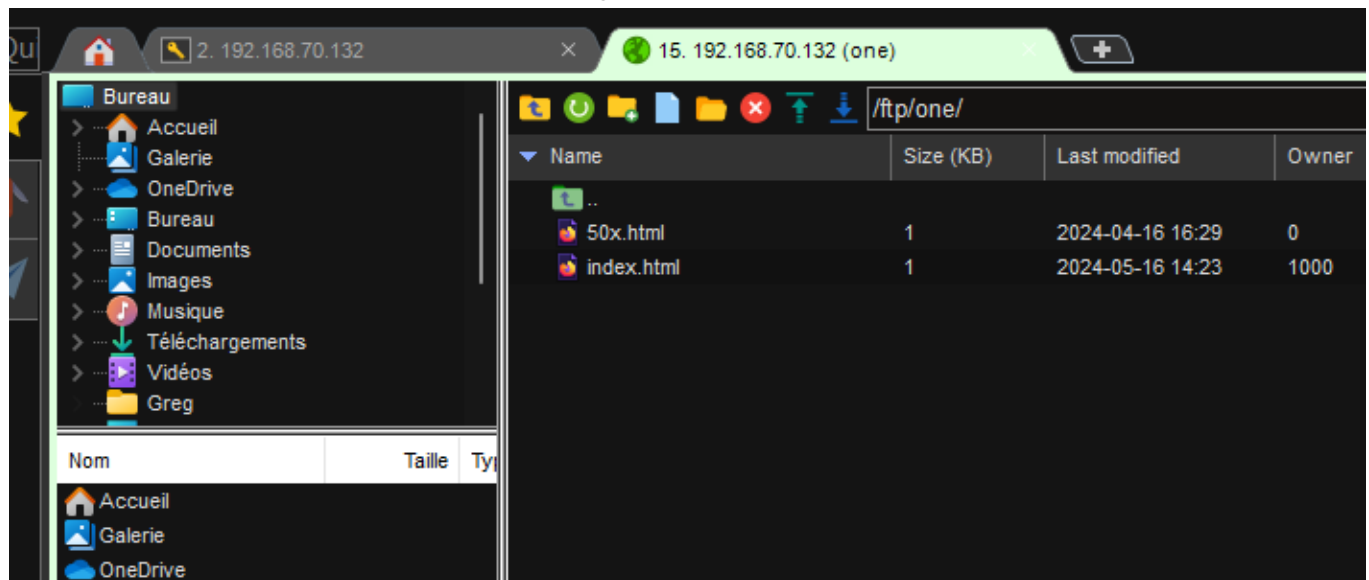
```
ftp:
  image: 'delfer/alpine-ftp-server:latest'
  #Configure le conteneur pour utiliser le réseau de
  l'hôte, ce qui signifie qu'il partage l'espace de noms
  réseau de l'hôte
  network_mode: host
  volumes:
    - web-data:/ftp/one
  #ports:
  # - '21:21'
  #- '21000-21010:21000-21010'
  environment:
    - 'USERS=one|123456789'
  #Restart le conteneur s'il s'arrête
  restart: always

volumes:
  #Volume qui est partagé entre mes services web et ftp. Il
  est utilisé pour stocker les fichiers que l'on souhaite se
  servir avec nginx et transférer avec FTP
  web-data:
```

Je lance ensuite mon docker-compose :

```
sudo docker-compose up -d
```

Mes conteneurs vont se lancer, je me connecte donc en FTP:



Je modifie donc l'index.html pour qu'apparaisse ce que je veux, puis me connecte avec :

```
192.168.70.132:8080/index.html
```

Et voilà !