

# Guide Général : Configuration d'un Load Balancer avec Docker Compose

## Introduction

Ce document fournit une méthode générale pour configurer un Load Balancer avec Docker Compose. Cette configuration est applicable à plusieurs types de services backend, pas uniquement PHP. Le Load Balancer distribue les requêtes entre plusieurs instances de services pour améliorer la disponibilité et l'équilibrage de la charge.

## 1. Fichier docker-compose.yml

Le fichier docker-compose.yml définit plusieurs services backend et un service Load Balancer. Voici un exemple :

```
version: '3.8'
```

```
services:
```

```
  backend1:
```

```
    build: ./backend
```

```
    container_name: backend1
```

```
    expose:
```

```
      - '80'
```

```
  backend2:
```

```
    build: ./backend
```

```
    container_name: backend2
```

```
    expose:
```

```
      - '80'
```

# Guide Général : Configuration d'un Load Balancer avec Docker Compose

load-balancer:

build:

context: ./load-balancer

ports:

- '9000:80'

## 2. Dockerfile des services backend

Chaque service backend est défini dans un Dockerfile. Voici un exemple pour un backend générique :

```
FROM <image de base, par exemple, node:14 ou php:7.4-apache>
```

```
WORKDIR /usr/src/app
```

```
COPY . .
```

```
RUN <commande de construction spécifique au service>
```

```
EXPOSE 80
```

```
CMD ["<commande de démarrage spécifique au service>"]
```

## 3. Dockerfile du Load Balancer

Le Load Balancer est configuré avec Apache. Voici un exemple de Dockerfile :

```
FROM debian:latest
```

```
RUN apt-get update && apt-get install apache2 -y
```

```
RUN a2enmod proxy proxy_http proxy_balancer lbmethod_byrequests
```

```
COPY vhost.conf /etc/apache2/sites-available/000-default.conf
```

# Guide Général : Configuration d'un Load Balancer avec Docker Compose

EXPOSE 80

CMD ["apachectl", "-D", "FOREGROUND"]

## 4. Configuration du Load Balancer

La configuration du Load Balancer est définie dans un fichier vhost.conf (ou similaire). Voici un exemple :

```
<VirtualHost *:80>

    <Proxy balancer://mycluster>

        BalancerMember http://backend1:80

        BalancerMember http://backend2:80

    </Proxy>

    ProxyPass / balancer://mycluster/

    ProxyPassReverse / balancer://mycluster/

</VirtualHost>
```

## 5. Étapes pour lancer les conteneurs

1. Construisez et démarrez les conteneurs :

```
docker compose up --build -d
```

2. Vérifiez que les conteneurs fonctionnent :

```
docker ps
```

3. Accédez au Load Balancer via un navigateur ou un outil comme curl :

```
http://localhost:9000
```

## **Guide Général : Configuration d'un Load Balancer avec Docker Compose**

Le Load Balancer distribuera les requêtes entre les services backend.

### **Conclusion**

Cette méthode est générale et peut être adaptée à divers types de services backend (Node.js, PHP, Python, etc.). Le Load Balancer Apache peut être remplacé par un autre outil comme Nginx en modifiant la configuration. Cette approche garantit une répartition efficace de la charge entre plusieurs instances de service.