

Mise en place d'un environnement Docker avec Traefik, Portainer, Watchtower et Pterodactyl

Mise en place d'un environnement Docker avec Traefik, Portainer, Watchtower et Pterodactyl	1
1. Introduction	2
2. Préparation de l'environnement	2
3. Installation de Docker et Docker Compose	2
4. Traefik : Reverse Proxy	3
5. Portainer : Interface graphique Docker	4
6. Watchtower : Mise à jour automatique	5
7. Pterodactyl : Panel de gestion de serveurs de jeux	5
8. Tests et vérifications	6
9. Sécurisation avancée	6
10. Sauvegarde et maintenance	6
11. Conclusion	7

1. Introduction

Cette documentation décrit l'installation et la configuration d'un environnement de conteneurisation Docker incluant les outils suivants :

- **Docker & Docker Compose** : système de conteneurisation
- **Traefik** : reverse proxy dynamique
- **Portainer** : interface graphique de gestion Docker
- **Watchtower** : mise à jour automatique des conteneurs
- **Pterodactyl** : gestion de serveurs de jeux

L'ensemble de cette infrastructure repose sur une machine virtuelle (VM) Debian ou Ubuntu, installée via **Proxmox**.

2. Préparation de l'environnement

2.1 Prérequis

- Une machine Debian 11/12 ou Ubuntu 22.04
- Droits administrateurs (root ou sudo)
- Connexion internet stable

2.2 Mise à jour du système

```
sudo apt update && sudo apt upgrade -y
```

3. Installation de Docker et Docker Compose

3.1 Installer les dépendances

```
sudo apt install -y ca-certificates curl gnupg lsb-release  
apt-transport-https software-properties-common
```

3.2 Ajouter le dépôt Docker

```
curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg  
--dearmor -o /usr/share/keyrings/docker.gpg  
echo \  
  "deb [arch=$(dpkg --print-architecture)  
signed-by=/usr/share/keyrings/docker.gpg]  
https://download.docker.com/linux/debian \  
  $(lsb_release -cs) stable" | sudo tee  
/etc/apt/sources.list.d/docker.list > /dev/null
```

3.3 Installer Docker

```
sudo apt update  
sudo apt install -y docker-ce docker-ce-cli containerd.io  
docker-compose-plugin  
sudo systemctl enable docker  
sudo systemctl start docker
```

4. Traefik : Reverse Proxy

4.1 Structure des fichiers

```
/docker/traefik/  
├── traefik.yml  
├── acme.json  
└── docker-compose.yml
```

4.2 Configuration de base : traefik.yml

```
entryPoints:  
  web:  
    address: ":80"  
  websecure:  
    address: ":443"  
  
providers:  
  docker:  
    exposedByDefault: false  
  
api:  
  dashboard: true  
  
certificatesResolvers:  
  myresolver:  
    selfSigned: {}
```

4.3 Sécurité : créer le fichier acme.json

```
touch acme.json  
chmod 600 acme.json
```

4.4 Déploiement : docker-compose.yml

```
version: '3.8'  
  
services:  
  traefik:  
    image: traefik:v2.10  
    container_name: traefik  
    restart: always  
    command:  
      - "--providers.docker=true"  
      - "--entrypoints.web.address=:80"  
      - "--entrypoints.websecure.address=:443"
```

```
- "--api.dashboard=true"
- "--certificatesresolvers.myresolver.selfSigned=true"
ports:
- "80:80"
- "443:443"
- "8080:8080"
volumes:
- "/var/run/docker.sock:/var/run/docker.sock:ro"
- "./traefik.yml:/traefik.yml:ro"
- "./acme.json:/acme.json"
```

5. Portainer : Interface graphique Docker

5.1 Création du volume persistant

`docker volume create portainer_data`

5.2 Lancement du conteneur

```
docker run -d \  
  --name=portainer \  
  --restart=always \  
  -p 9000:9000 \  
  -v /var/run/docker.sock:/var/run/docker.sock \  
  -v portainer_data:/data \  
  portainer/portainer-ce
```

5.3 Intégration avec Traefik (extrait docker-compose)

```
labels:  
  - "traefik.enable=true"  
  - "traefik.http.routers.portainer.rule=Host(`portainer.local`)">  
  - "traefik.http.routers.portainer.entrypoints=websecure"  
  - "traefik.http.services.portainer.loadbalancer.server.port=9000"
```

6. Watchtower : Mise à jour automatique

6.1 Déploiement simple

```
docker run -d \  
  --name watchtower \  
  --restart=always \  
  -v /var/run/docker.sock:/var/run/docker.sock \  
  containrrr/watchtower \  
  --cleanup --interval 3600
```

6.2 Vérification

`docker logs watchtower`

7. Pterodactyl : Panel de gestion de serveurs de jeux

7.1 Installation du panel

- Nécessite PHP, MySQL/MariaDB, Redis, Node.js
- Le panel est installé en dehors de Docker, sur une machine dédiée

```
cd /var/www
git clone https://github.com/pterodactyl/panel.git
cd panel
cp .env.example .env
composer install --no-dev --optimize-autoloader
php artisan key:generate
php artisan p:environment:setup
php artisan migrate --seed
```

7.2 Déploiement de Wings via Docker

```
docker run -d \
  --name wings \
  --restart=always \
  --network host \
  -v /etc/pterodactyl:/etc/pterodactyl \
  -v /var/lib/pterodactyl:/var/lib/pterodactyl \
  ghcr.io/pterodactyl/wings:latest
```

8. Tests et vérifications

Service	URL	État attendu
Docker	<code>docker ps</code>	Liste des conteneurs
Traefik	<code>http://IP:8080</code>	Dashboard visible
Portainer	<code>http://IP:9000</code>	Interface Web accessible
Watchtower	<code>docker logs watchtower</code>	Mises à jour détectées
Pterodactyl	<code>http://panel.domaine. local</code>	Interface Web panel

9. Sécurisation avancée

9.1 Passer en HTTPS avec Traefik

- Traefik permet le HTTPS via Let's Encrypt ou un certificat local
- Pour Let's Encrypt : ajouter dans `traefik.yml`

```
certificatesResolvers:  
  letsencrypt:  
    acme:  
      email: votremail@example.com  
      storage: acme.json  
      httpChallenge:  
        entryPoint: web
```

9.2 Générer un certificat local (optionnel)

```
openssl genrsa -out ca.key 4096  
openssl req -x509 -new -nodes -key ca.key -sha256 -days 3650 -out ca.crt
```

10. Sauvegarde et maintenance

10.1 Sauvegarde Proxmox

- Sauvegarder les VM contenant Docker via l'interface Proxmox
- Utiliser les **snapshots** pour les points de restauration

10.2 Scripts de backup Docker (ex : docker-backup.sh)

```
#!/bin/bash
DATE=$(date +"%Y-%m-%d")
BACKUP_DIR=/opt/backup/$DATE
mkdir -p $BACKUP_DIR
docker ps -q | while read container; do
    NAME=$(docker inspect --format='{{.Name}}' $container | sed 's/\///')
    docker export $container > "$BACKUP_DIR/$NAME.tar"
done
```

11. Conclusion

Ce guide fournit une base robuste pour créer un environnement de conteneurisation complet, sécurisé et automatisé. Traefik assure le routage, Portainer la gestion graphique, Watchtower la maintenance, et Pterodactyl permet l'hébergement de serveurs de jeux. Il est recommandé de combiner cette infrastructure avec un pare-feu (comme OPNsense) et un outil de supervision comme GLPI ou Zabbix pour une solution complète.