

# Guide de Déploiement en Production

Ce guide explique comment déployer l'API en production sur un serveur Linux.

## Prérequis

- Serveur Linux (Ubuntu 20.04+ recommandé)
- Python 3.8+
- MySQL/MariaDB
- Nginx
- Accès root ou sudo

## 1. Préparation du Serveur

### Mise à jour du système

```
bash
```

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

### Installation des dépendances

```
bash
```

```
# Python et pip
```

```
sudo apt install python3 python3-pip python3-venv -y
```

```
# MySQL
```

```
sudo apt install mysql-server -y
```

```
# Nginx
```

```
sudo apt install nginx -y
```

```
# Outils supplémentaires
```

```
sudo apt install git supervisor -y
```

## 2. Configuration MySQL

### Sécuriser MySQL

```
bash  
  
sudo mysql_secure_installation
```

### Créer la base de données et l'utilisateur

```
bash  
  
sudo mysql -u root -p
```

```
sql  
  
-- Créer la base de données  
CREATE DATABASE trait8 CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;  
  
-- Créer un utilisateur dédié  
CREATE USER 'trait8_user'@'localhost' IDENTIFIED BY 'VOTRE_MOT_DE_PASSE_SECURISE';  
  
-- Donner les permissions  
GRANT ALL PRIVILEGES ON trait8.* TO 'trait8_user'@'localhost';  
FLUSH PRIVILEGES;  
  
EXIT;
```

### Importer le schéma

```
bash  
  
mysql -u trait8_user -p trait8 < database_corrections.sql
```

## 3. Déploiement de l'Application

### Créer un utilisateur dédié

```
bash  
  
sudo useradd -m -s /bin/bash trait8  
sudo su - trait8
```

## Cloner/Copier le projet

```
bash

cd ~
mkdir api_materiels
cd api_materiels

# Copier vos fichiers ici ou cloner depuis git
# git clone <votre-repo>
```

## Créer un environnement virtuel

```
bash

python3 -m venv venv
source venv/bin/activate
```

## Installer les dépendances

```
bash

pip install -r requirements.txt
pip install gunicorn
```

## Configuration des variables d'environnement

```
bash

nano .env
```

Contenu du fichier `.env`:

```
env
```

```
DB_HOST=localhost
DB_USER=trait8_user
DB_PASSWORD=VOTRE_MOT_DE_PASSE_SECURISE
DB_NAME=trait8
DB_CHARSET=utf8mb4

SECRET_KEY=GENERER_UNE_CLE_SECRETE_TRES_LONGUE_ET_ALEATOIRE
ALGORITHM=HS256
ACCESS_TOKEN_EXPIRE_MINUTES=1440

ALLOWED_ORIGINS=https://votre-domaine.com

DEBUG=False
```

## Modifier config/database.py pour utiliser les variables d'environnement

```
python

import os
from dotenv import load_dotenv

load_dotenv()

class Database:
    @staticmethod
    def get_connection():
        try:
            connection = mysql.connector.connect(
                host=os.getenv('DB_HOST', 'localhost'),
                user=os.getenv('DB_USER', 'root'),
                password=os.getenv('DB_PASSWORD', ''),
                database=os.getenv('DB_NAME', 'trait8'),
                charset=os.getenv('DB_CHARSET', 'utf8mb4'),
                use_unicode=True
            )
            return connection
        except Error as e:
            print(f"Erreur de connexion à la base de données: {e}")
            raise
```

## Créer les dossiers nécessaires

```
bash  
mkdir -p uploads logs
```

## Tester l'application

```
bash  
  
gunicorn main:app --bind 0.0.0.0:8000 --workers 2 --timeout 120  
# Ctrl+C pour arrêter
```

## 4. Configuration Supervisor

Supervisor permet de gérer l'application comme un service.

```
bash  
  
exit # Sortir de l'utilisateur trait8  
sudo nano /etc/supervisor/conf.d/trait8_api.conf
```

Contenu:

```
ini  
  
[program:trait8_api]  
directory=/home/trait8/api_materiels  
command=/home/trait8/api_materiels/venv/bin/gunicorn main:app --bind 127.0.0.1:8000 --workers 4 --worker-class uvicorn.  
user=trait8  
autostart=true  
autorestart=true  
stderr_logfile=/var/log/trait8_api.err.log  
stdout_logfile=/var/log/trait8_api.out.log  
environment=PATH="/home/trait8/api_materiels/venv/bin"
```

## Activer et démarrer

```
bash
```

```
sudo supervisorctl reread
sudo supervisorctl update
sudo supervisorctl start trait8_api
sudo supervisorctl status trait8_api
```

## 5. Configuration Nginx

### Créer la configuration

```
bash

sudo nano /etc/nginx/sites-available/trait8_api
```

Contenu:

```
nginx

server {
    listen 80;
    server_name api.votre-domaine.com; # Remplacer par votre domaine

    client_max_body_size 20M; # Pour permettre l'upload de fichiers

    location / {
        proxy_pass http://127.0.0.1:8000;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
        proxy_redirect off;

        # Timeouts
        proxy_connect_timeout 120s;
        proxy_send_timeout 120s;
        proxy_read_timeout 120s;
    }

    # Logs
    access_log /var/log/nginx/trait8_api_access.log;
    error_log /var/log/nginx/trait8_api_error.log;
}
```

## Activer le site

```
bash

sudo ln -s /etc/nginx/sites-available/trait8_api /etc/nginx/sites-enabled/
sudo nginx -t # Tester la configuration
sudo systemctl restart nginx
```

## 6. Sécurisation avec HTTPS (Certbot)

### Installer Certbot

```
bash

sudo apt install certbot python3-certbot-nginx -y
```

### Obtenir un certificat SSL

```
bash

sudo certbot --nginx -d api.votre-domaine.com
```

Suivre les instructions. Certbot configurera automatiquement Nginx pour HTTPS.

### Renouvellement automatique

```
bash

sudo certbot renew --dry-run # Tester
# Le renouvellement automatique est configuré par défaut
```

## 7. Firewall

### Configurer UFW

```
bash

sudo ufw allow 22/tcp # SSH
sudo ufw allow 80/tcp # HTTP
sudo ufw allow 443/tcp # HTTPS
sudo ufw enable
sudo ufw status
```

## 8. Sauvegardes Automatiques

### Script de sauvegarde

```
bash

sudo nano /usr/local/bin/backup_trait8.sh
```

Contenu:

```
bash

#!/bin/bash

# Variables
DATE=$(date +%Y%m%d_%H%M%S)
BACKUP_DIR="/home/trait8/backups"
DB_NAME="trait8"
DB_USER="trait8_user"
DB_PASS="VOTRE_MOT_DE_PASSE"

# Créer le dossier de sauvegarde
mkdir -p $BACKUP_DIR

# Sauvegarder la base de données
mysqldump -u $DB_USER -p$DB_PASS $DB_NAME | gzip > $BACKUP_DIR/db_backup_$DATE.sql.gz

# Garder seulement les 7 dernières sauvegardes
find $BACKUP_DIR -name "db_backup_*.sql.gz" -mtime +7 -delete

echo "Sauvegarde terminée: $DATE"
```

### Rendre exécutable

```
bash

sudo chmod +x /usr/local/bin/backup_trait8.sh
```

### Ajouter au cron (tous les jours à 2h du matin)

```
bash

sudo crontab -e
```



Ajouter:

```
0 2 * * * /usr/local/bin/backup_trait8.sh >> /var/log/trait8_backup.log 2>&1
```

## 9. Monitoring

### Installer monitoring basique

```
bash  
  
sudo apt install htop iotop nethogs -y
```

### Logs à surveiller

```
bash  
  
# Logs de l'application  
tail -f /home/trait8/api_materiels/logs/error.log  
  
# Logs Nginx  
tail -f /var/log/nginx/trait8_api_error.log  
  
# Logs Supervisor  
tail -f /var/log/trait8_api.err.log
```

## 10. Commandes Utiles

### Redémarrer l'application

```
bash  
  
sudo supervisorctl restart trait8_api
```

### Voir les logs en direct

```
bash  
  
sudo supervisorctl tail -f trait8_api
```

### Redémarrer Nginx

```
bash
```

```
sudo systemctl restart nginx
```

## Vérifier l'état

```
bash
```

```
sudo supervisorctl status
```

```
sudo systemctl status nginx
```

```
sudo systemctl status mysql
```

## Mise à jour de l'application

```
bash
```

```
sudo su - trait8
```

```
cd ~/api_materiels
```

```
source venv/bin/activate
```

```
git pull # ou copier les nouveaux fichiers
```

```
pip install -r requirements.txt
```

```
exit
```

```
sudo supervisorctl restart trait8_api
```

## 11. Sécurité Supplémentaire

### Fail2Ban (protection contre les attaques)

```
bash
```

```
sudo apt install fail2ban -y
```

```
sudo systemctl enable fail2ban
```

```
sudo systemctl start fail2ban
```

### Limiter les connexions SSH

```
bash
```

```
sudo nano /etc/ssh/sshd_config
```

Modifier:

```
PermitRootLogin no
```

```
PasswordAuthentication no # Si vous utilisez des clés SSH
```

```
MaxAuthTries 3
```

```
bash
```

```
sudo systemctl restart sshd
```

## 12. Checklist de Production

- ☐ Base de données créée et sécurisée
- ☐ Utilisateur dédié créé
- ☐ Application déployée avec environnement virtuel
- ☐ Variables d'environnement configurées
- ☐ Supervisor configuré et fonctionnel
- ☐ Nginx configuré
- ☐ HTTPS activé (SSL)
- ☐ Firewall configuré
- ☐ Sauvegardes automatiques en place
- ☐ Logs accessibles et surveillés
- ☐ Documentation mise à jour

## Dépannage

### L'application ne démarre pas

```
bash
```

```
# Vérifier les logs
```

```
sudo supervisorctl tail -f trait8_api stderr
```

```
# Vérifier la configuration
```

```
sudo supervisorctl reread
```

```
sudo supervisorctl update
```

### Erreurs de base de données

```
bash
```

```
# Vérifier la connexion
```

```
mysql -u trait8_user -p trait8
```

```
# Vérifier les permissions
```

```
SHOW GRANTS FOR 'trait8_user'@'localhost';
```

## Problèmes Nginx

```
bash
```

```
# Tester la configuration
```

```
sudo nginx -t
```

```
# Voir les erreurs
```

```
sudo tail -f /var/log/nginx/error.log
```

## Support

Pour toute question ou problème, consultez les logs et la documentation de l'API.