

Votre pile web Ubuntu prête : Nginx, PHP-FPM et MariaDB

Suivez cette checklist pour installer, relier et contrôler les composants essentiels d'une pile LEMP sur Ubuntu 22.04. Cochez chaque validation avant de passer à l'hébergement d'une application.

Au sommaire

1. Préparer le serveur
2. Installer et vérifier les services
3. Repères : qui fait quoi ?
4. 3. Relier Nginx à PHP-FPM
5. 4. Tester avant de déployer une application

1. Préparer le serveur

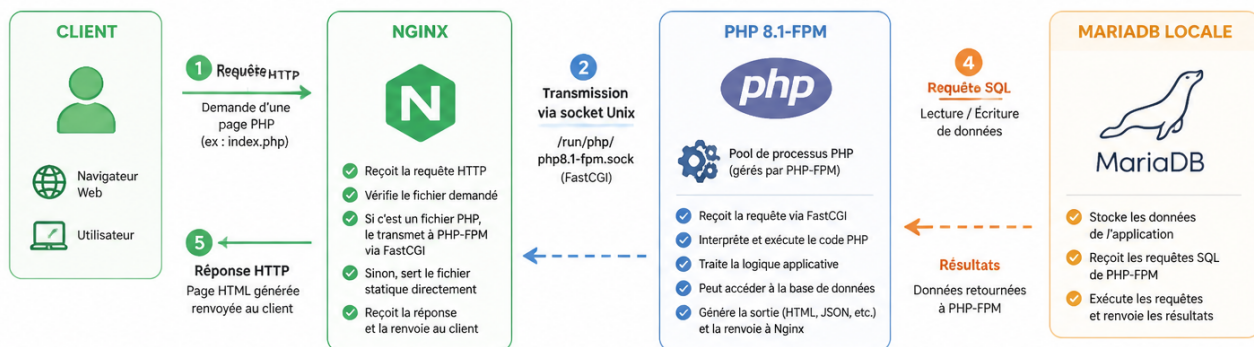
- ☐ **Utiliser Ubuntu 22.04 LTS amd64 avec un utilisateur non-root disposant de sudo.**
Vérifier les privilèges avec `sudo -v`.
- ☐ **Conserver une session SSH ouverte avant toute modification du pare-feu.**
Contrôler l'état avec `sudo ufw status`.
- ☐ **Vérifier la mémoire disponible.**
Utiliser `free -m` avant d'ajuster PHP-FPM ou MariaDB.
- ☐ **Mettre à jour les paquets système.**
Exécuter `sudo apt update && sudo apt upgrade -y`, puis vérifier la version avec `lsb_release -a`.

TRAITEMENT D'UNE REQUÊTE PHP AVEC NGINX, PHP-FPM ET MARIADB

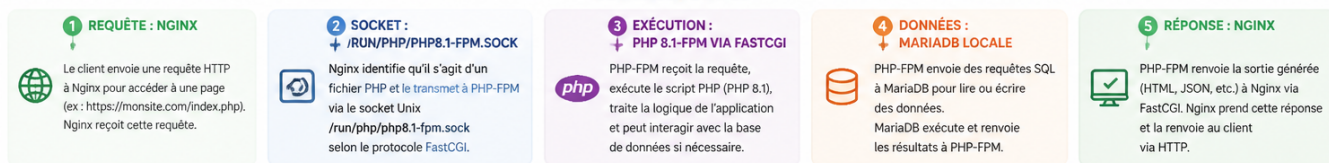


Lorsqu'un utilisateur demande une page PHP, Nginx sert de point d'entrée, transmet la requête à PHP-FPM pour exécution, qui peut interagir avec MariaDB pour récupérer ou enregistrer des données, puis renvoie la réponse à Nginx qui la retourne au client.

- 1 REQUÊTE : NGINX
- 2 SOCKET : /RUN/PHP/PHP8.1-FPM.SOCK
- 3 EXÉCUTION : PHP 8.1-FPM VIA FASTCGI
- 4 DONNÉES : MARIADB LOCALE
- 5 RÉPONSE : NGINX



DÉTAIL DES ÉTAPES



À RETENIR

- Nginx = serveur web performant et léger
- PHP-FPM = exécution du code PHP de façon isolée et scalable
- MariaDB = base de données relationnelle fiable et rapide
- La communication Nginx ↔ PHP-FPM se fait via FastCGI sur socket Unix
- L'ensemble fonctionne en local pour de meilleures performances et sécurité

ENVIRONNEMENT TYPIQUE



AVANTAGES

- ✓ Haute performance
- ✓ Faible consommation
- ✓ Sécurité renforcée
- ✓ Architecture modulaire et maintenable

INFOGRAPHIE Schéma de configuration FastCGI entre Nginx, PHP-FPM et MariaDB sur Ubuntu 22.04.

2. Installer et vérifier les services

Installer Nginx, le démarrer et l'activer au démarrage.

Vérifier avec `sudo systemctl status nginx --no-pager`.

Si UFW est actif, autoriser le profil HTTP/HTTPS de Nginx.

Utiliser `sudo ufw allow 'Nginx Full'` sans retirer la règle SSH.

Installer MariaDB, le démarrer et l'activer au démarrage.

Lancer ensuite `sudo mysql_secure_installation`.

Installer PHP 8.1-FPM et les modules requis par les applications PHP.

Inclure au minimum `php-mysql` ; contrôler PHP avec `php -v` et le service avec `systemctl`.

Repères : qui fait quoi ?

Nginx reçoit les requêtes HTTP, sert les fichiers statiques et transmet les scripts PHP à PHP-FPM.

MariaDB stocke les données de l'application et doit rester accessible localement plutôt qu'exposée à Internet.

PHP-FPM exécute le code PHP ; Nginx ne traite pas directement les fichiers PHP.

La liaison clé est le socket Unix PHP-FPM : `/run/php/php8.1-fpm.sock`.

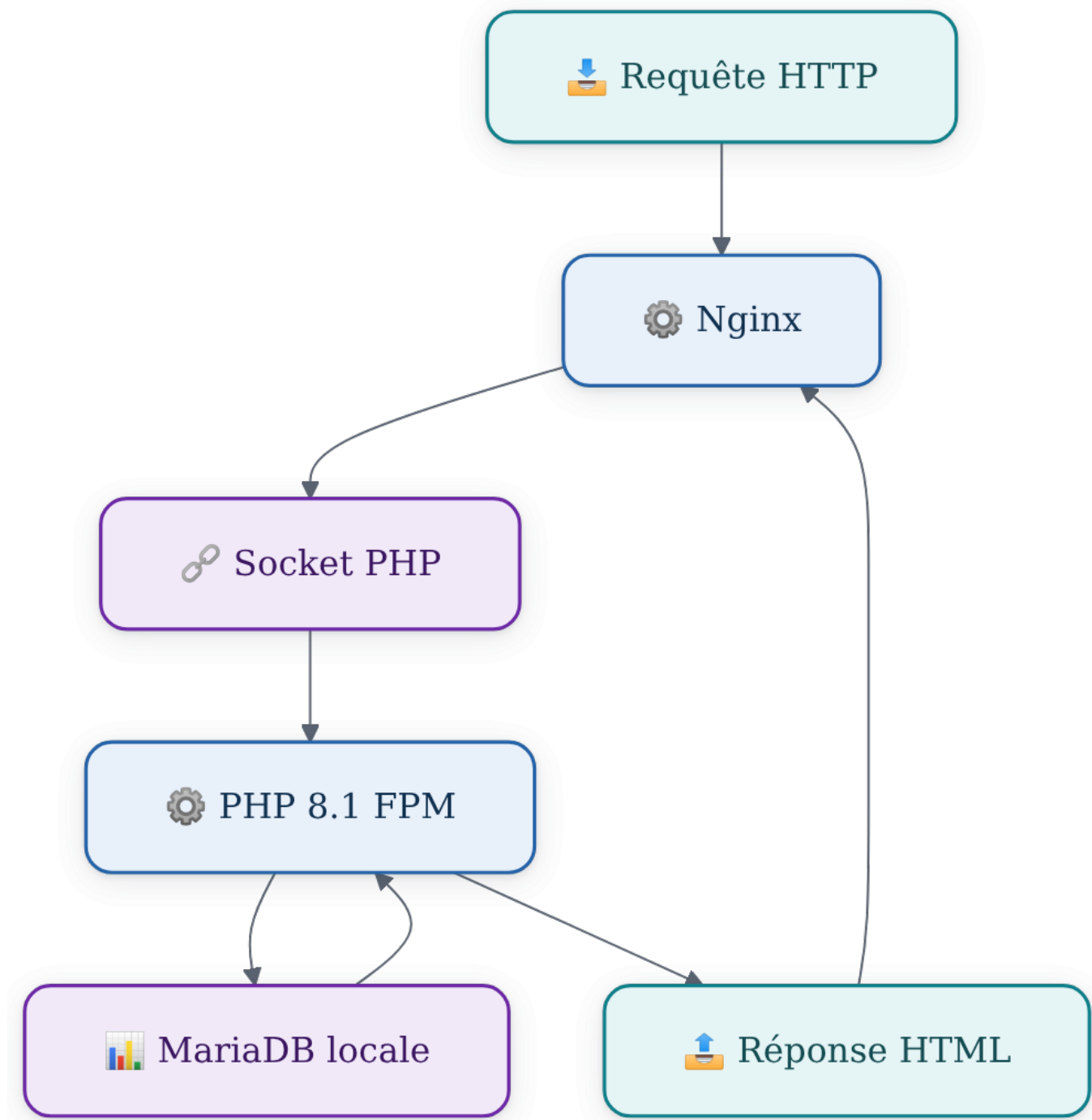
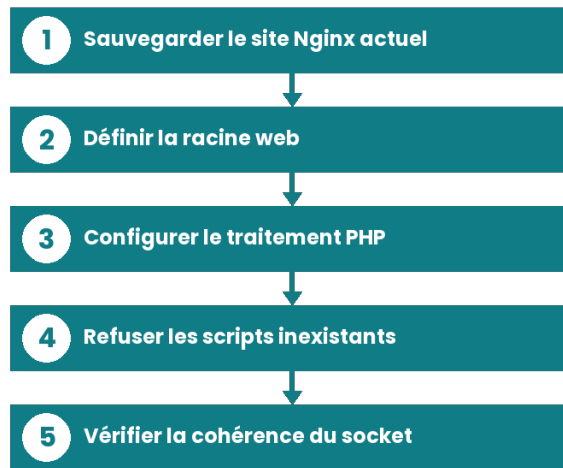


SCHÉMA Circulation d'une requête entre Nginx, PHP-FPM et MariaDB sur le serveur Ubuntu.

3. Relier Nginx à PHP-FPM



1 **Sauvegarder le site Nginx actuel**

Conserver une copie de la configuration par défaut avant de la remplacer afin de pouvoir revenir en arrière.

2 **Définir la racine web**

Utiliser `/var/www/html` pour le site de test décrit dans le guide.

3 **Configurer le traitement PHP**

Ajouter le bloc de traitement des fichiers `.php`, inclure la configuration FastCGI d'Ubuntu et transmettre les scripts au socket PHP-FPM.

4 **Refuser les scripts inexistants**

Prévoir le contrôle des fichiers PHP demandés pour éviter les erreurs de routage et l'exposition de ressources inattendues.

5 **Vérifier la cohérence du socket**

Le chemin déclaré dans Nginx doit correspondre exactement au socket du service PHP 8.1-FPM : `/run/php/php8.1-fpm.sock`.

4. Tester avant de déployer une application

- ☐ **Vérifier que Nginx est actif.**
La sortie doit indiquer `active (running)`.
- ☐ **Vérifier que PHP-FPM est actif et que la version PHP commence par 8.1.**
Contrôler le service `php8.1-fpm` et la commande `php -v`.
- ☐ **Tester la réponse locale de MariaDB.**
Exécuter `sudo mariadb` ; l'invite `MariaDB [(none)]>` confirme que le serveur répond.
- ☐ **Créer temporairement une page PHP de test pour confirmer l'exécution via Nginx.**
Utiliser ce test uniquement pour valider la chaîne Nginx → PHP-FPM.
- ☐ **Supprimer immédiatement le fichier de test après validation.**
Une page utilisant `phpinfo()` révèle des informations techniques sur le serveur.

! Contrôles de sécurité essentiels

Ne rendez pas MariaDB accessible publiquement par défaut. Une application PHP présente sur le même serveur doit se connecter via localhost avec un compte limité à sa propre base. Évitez aussi d'installer des paquets depuis des scripts téléchargés puis exécutés directement dans le terminal.

Cette fiche cible Ubuntu 22.04 LTS avec PHP 8.1-FPM. Si vous changez de version de PHP, adaptez également le nom du service et le chemin du socket PHP-FPM.