

Configuration complète d'un serveur LAMP sous UBUNTU 24.04

v1.0.0 | 20/01/2025 | Auteur : Bauer Baptiste

Chapitre | Durée de réalisation : 4

Table des matières

1. Etape 1 : Mise à jour du système	2
2. Étape 2 : Installation Apache, PHP et MySQL	3
3. Etape 3 : Configuration avancée d'Apache pour un projet Symfony et un site web classique	4
4. Etape 4 : Configuration des VirtualHost pour héberger plusieurs sites web	5
5. Etape 5 : Droit d'accès aux fichiers et répertoires	6
6. Etape 6 : Configuration de MySQL	7
7. Etape 7 : Installation de phpMyAdmin	8
8. Etape 8 : Installation et configuration de NodeJS	9
9. Etape 9 : Installation de MongoDB	10
10. Etape 10 : Sécurisation du serveur Linux	11
11. Etape 11 : Création service FTP	12

Compétences ciblées

- ☐ Installer et configurer un serveur web (Apache).
- ☐ Installer et configurer PHP avec Apache.
- ☐ Installer et sécuriser MySQL.
- ☐ Installer phpMyAdmin pour administrer MySQL via une interface web.
- ☐ Installer et configurer NodeJS.
- ☐ Installer MongoDB et créer une base de données simple.
- ☐ Sécuriser un serveur Linux (firewall, gestion des permissions).
- ☐ Créer et gérer des utilisateurs FTP avec accès restreint.

1. Etape 1 : Mise à jour du système

Pour garantir le bon fonctionnement de votre machine virtuelle, il est essentiel de procéder à une mise à jour régulière des paquets logiciels. Cette opération permet de bénéficier des dernières corrections de bugs et des améliorations de sécurité.

Q1) Quelles commandes permettent de mettre à jour l'ensemble des paquets logiciels d'une machine Linux ?

réponse 1 disponible.

2. Étape 2 : Installation Apache, PHP et MySQL

Pour configurer un serveur web complet, il est nécessaire d'installer les composants suivants :

- Apache : serveur web open source, largement utilisé pour héberger des sites web.
- PHP : langage de programmation côté serveur, indispensable pour la création de sites dynamiques.
- MySQL : système de gestion de base de données relationnelle, permettant de stocker et d'interroger des données.

Q2) Quelles commandes permettent d'installer Apache, PHP et MySQL sur une machine Linux ? Vérifier ensuite l'installation d'apache en ouvrant un navigateur web et en saisissant l'adresse IP de la machine virtuelle.

réponse 2 disponible.

3. Etape 3 : Configuration avancée d'Apache pour un projet Symfony et un site web classique

Pour personnaliser la configuration d'Apache, il est nécessaire de modifier certains fichiers de configuration. Ces fichiers sont situés dans le répertoire `/etc/apache2/` ou d'activer des modules spécifiques.

Q3) Quelles sont les modules Apache nécessaires pour exécuter un projet Symfony ?
Comment les activer ?

réponse 3 disponible.

4. Etape 4 : Configuration des VirtualHost pour héberger plusieurs sites web

Nous pouvons modifier les fichiers appelés "VirtualHost" pour configurer les paramètres spécifiques à chaque site web hébergé sur le serveur Apache.

Imaginons que nous ayons deux sites web à héberger sur notre serveur :

- `projet1.symfony.3wa` : Projet 1 développé avec Symfony.
- `projet2.legacy.3wa` : Projet 2 développé avec PHP et HTML sans framework.

Q4) Comment configurer les **VirtualHost** pour héberger ces deux sites web sur un même serveur Apache ?

réponse 4 disponible.

5. Etape 5 : Droit d'accès aux fichiers et répertoires

Pour des raisons de sécurité, il est essentiel de définir les bons droits d'accès aux fichiers et répertoires de votre serveur web. Cela permet de limiter les risques.

Assurez-vous que le serveur web peut accéder aux fichiers de cache et log de Symfony.

Q5) Quels droits d'accès sont recommandés pour les fichiers et répertoires d'un projet Symfony ?

Quelle commande permet de les appliquer ?

réponse 5 disponible.

6. Etape 6 : Configuration de MySQL

Pour sécuriser votre serveur MySQL, il est recommandé de définir un mot de passe pour l'utilisateur `root` et de supprimer les utilisateurs inutiles.

Q6) Comment définir un mot de passe pour l'utilisateur `root` de MySQL ? Comment supprimer un utilisateur inutile ?

réponse 6 disponible.

7. Etape 7 : Installation de phpMyAdmin

phpMyAdmin est une interface web permettant de gérer les bases de données MySQL de manière graphique. Cela facilite la création, la modification et la suppression de bases de données et de tables.

Q7) Comment installer phpMyAdmin sur une machine Linux ? Comment y accéder via un navigateur web ?

réponse 7 disponible.

8. Etape 8 : Installation et configuration de NodeJS

Node.js est un environnement d'exécution JavaScript côté serveur, basé sur le moteur JavaScript V8 de Google Chrome. Il permet d'exécuter du code JavaScript côté serveur.

Q8) Comment installer NodeJS sur une machine Linux ? Comment vérifier l'installation de NodeJS et de npm ?

réponse 8 disponible.

9. Etape 9 : Installation de MongoDB

MongoDB est une base de données NoSQL orientée document, qui stocke les données sous forme de documents JSON. Elle est très utilisée pour les applications web modernes.

Q9) Comment installer MongoDB sur une machine Linux ? Comment vérifier que MongoDB est en cours d'exécution

réponse 9 disponible.

10. Etape 10 : Sécurisation du serveur Linux

Pour sécuriser votre serveur Linux, il est essentiel de mettre en place un pare-feu et de gérer les permissions des utilisateurs.

Q10) Comment configurer un pare-feu sur une machine Linux ? Comment gérer les permissions des utilisateurs pour renforcer la sécurité du serveur ? Lister les services à autoriser dans le pare-feu pour un serveur web comme celui que nous avons configuré.

réponse 10 disponible.

Q11) Comment sécuriser au mieux l'accès SSH à votre serveur Linux ?

réponse 11 disponible.

Q12) Quand trop de tentative de connexion échoue, comment bloquer l'adresse IP de l'attaquant ?

réponse 12 disponible.

11. Etape 11 : Création service FTP

Q13) Sur Ubuntu, nous utiliserons le service FTP **SFTP**. Comment installer SFTP et vérifier que le service fonctionne ?

réponse 13 disponible.

Q14) Comment créer un groupe et un utilisateur SFTP sécurisé ?

réponse 14 disponible.

Q15) Donner les droits corrects sur le dossier web afin de permettre à Apache et SFTP d'accéder au dossier web.

réponse 15 disponible.

Q16) Modifier la configuration **SSH** pour pointer vers **/var/www**

réponse 16 disponible.