

Mise en production d'applications sans conteneurisation

Modalités

- Travail individuel / Binôme en autonomie
- Durée : 8 heures (suggéré comme une journée complète ou 2 demi-journées).
- Prérequis : Connaissances basiques des systèmes d'exploitation basés sur Unix/Linux, notions de base en réseau
- Outils requis : Machine sous Linux (Ubuntu)

Objectifs de l'activité

Découvrir la méthode de mise en production d'un site web de manière non conteneurisées et comprendre les limites de cette approche

Compétences

- Installer des packages sur Ubuntu
- Configuration de Nginx pour du reverse proxy
- Configuration d'un serveur web PHP afin que votre site web soit accessible depuis l'extérieur

Consignes

1 — Mise en situation

Vous venez d'être embauché dans l'entreprise TechFlow, une entreprise spécialisée dans le développement d'applications web. Elle a su s'imposer comme une référence dans le milieu par la qualité de leurs projets. Pour le développement de leur application, ils utilisent le langage de programmation PHP de manière native (ps: c'est un exemple pour l'exercice, dans le monde pro, une personne raisonnable ne fait pas ça ;)).

Pour le déploiement de leur application, ils utilisent un serveur Ubuntu et sur lequel ils utilisent PHP-FPM et Nginx pour pouvoir exposer leurs applications.

Livrables

- Faites une recherche sur PHP-FPM et NGINX. Expliquez l'utilité des différents outils
- Faites un schéma de l'interaction entre PHP-FPM, Nginx et votre code source quand un utilisateur souhaite accéder à la page d'accueil du site web

2 — Déploiement d'une application web PHP 7

L'année 2021 fut une année charnière pour TechFlow. Inspirée par le bourdonnement du marché et le besoin croissant de solutions digitales, l'équipe de développement avait une vision : créer DataHive, une plateforme PHP 7.4 destinée à stocker et analyser d'énormes quantités de données.

L'objectif de cette partie est de mettre en production le code source situé à l'adresse <https://github.com/WilliamBurillon/devops-training-php-7> sur le serveur qui est à votre disposition.

Pour cela, vous allez devoir :

- Installer php7.4 et php7.4-fpm sur la machine
- Installer Nginx
- Télécharger le code du repo précédent sur votre machine dans le dossier qui conventionnellement utilisé pour héberger le code source d'un site web
- Configurer Nginx pour qu'il puisse écouter sur le port 80 de votre machine, qu'il est accès à votre code source et qu'il utilise la socket php 7.4

Une fois cela fait, rendez vous à l'adresse suivante : `http://<ip-de-votre-machine>:80` et vous devriez voir site web. Votre site web est donc accessible depuis tout internet.

Livrables

- Avoir un site web fonctionnelle sur votre serveur sur le port 80

3 — Déploiement d'une application web PHP 8

Quelques années ont passé depuis le lancement réussi de DataHive. La technologie et les besoins du marché ont évolué. TechFlow a repéré une autre niche du marché et a décidé de développer une nouvelle application, InfoNest, basée sur PHP 8. Cependant, malgré le temps qui s'est écoulé, DataHive est toujours en opération et largement utilisé par sa clientèle, ne permettant pas à l'équipe technique de la mettre à jour.

L'objectif de cette partie est de mettre en production le code source situé à l'adresse <https://github.com/WilliamBurillon/devops-training-php-8> sur le serveur qui est à votre disposition. La différence avec la première partie est de pouvoir faire coexister deux projets n'utilisant pas la même version de PHP sur la même machine.

Vous allez devoir reprendre les différentes étapes précédentes, mais pour php8.0. Votre code source devra être dans un dossier différent du premier projet et avoir une autre configuration Nginx qui utilisera la bonne version de php et écoute sur le port 81 de votre machine.

Livrables

- Avoir le site Web 1 qui fonctionne sur le port 80 de votre machine
- Avoir le site Web 2 qui fonctionne sur le port 81 de votre machine

4 — Limite de cette approche

TechFlow a décroché énormément de contrat de prestation et va avoir énormément d'applications à développer et à déployer sur leur serveur. La problématique de cette nouvelle expansion réside dans la diversité des différentes technologies qui vont être utilisées (nodeJS, Python, GO, Java...).

Livrables

- Quelles sont les problématiques qui vont survenir par rapport à la gestion des différents code source (notamment par rapport à la diversité des technologies) ?

