## Exercices 5 – Nginx

Informations

**Évaluation** : formative

**Type de travail** : individuel

**Durée** : 2 heures

**Système d’exploitation** : Ubuntu serveur

**Environnement** : virtuel

Objectifs

Cet exercice a pour objectifs :

* Installer le serveur Web Nginx.
* Faire une configuration de base et avec des hôtes virtuels.

Dans cet exercice, vous allez installer un environnement de serveur Web (HTTP) nginx, regarder à sa configuration de base et configurer des hôtes virtuels.

**Note** : si vous fermez votre VM, n’oubliez pas d’arrêter vos conteneurs :

docker container stop *nom\_conteneur*

docker container start *nom\_conteneur*

#### Partie 1 : Installation de nginx

Dans cette partie, vous allez installer le serveur Web nginx selon les spécifications données.

Fichier de configuration par défaut  nginx.conf

Lancer nginx avec configuration personnalisé : nginx -c my\_nginx.conf

Étape 1 : Installation

1. Pour débuter, allez à l’image officielle sur le site de docker hub et prenez en note la commande pour lancer le conteneur.
2. Avant de lancer notre serveur, on va se créer un répertoire pour nos fichiers HTML de ce projet. À partir du répertoire Dev, créer un répertoire SiteNginx/html. Vous devriez avoir la structure suivante :

/home/user01/DevEx05/SiteNginx/html

1. Copiez le fichier index.html, que vous avez créé précédemment, dans html. Adapter le contenu au nouveau serveur.
2. Entrez dans le répertoire SiteNginx et exécutez la commande recommandée pour lancer l’image nginx. Ajuster la commande pour que le répertoire html soit monté sur le répertoire de documents du conteneur, en lecteur-écriture, et que le port 80 soit redirigé vers le port 8080.

*docker run --name some-nginx -v /home/user01/DevEx05/SiteNginx/html:/usr/share/nginx/html:rw -d nginx -p 8080:80*

*docker run --name some-nginx -p 8080:80 -v ‘’ $PWD ‘’:/usr/share/nginx/html:rw  -d nginx*

1. Vérifiez que votre image s’exécute bien.

docker container ls

ss -tnap | grep 8080

Ouvrez un navigateur et allez au site Web http://www.monsite.com:8080.

Avez-vous une page Web? Sinon, pourquoi?

Oui avec l’erreur 403



#### Partie 2 : Configurations

Dans cette partie, vous allez regarder la configuration par défaut et récupérer le fichier de configuration par défaut.

Étape 1 : Consultation des configurations par défaut

1. Connectez-vous sur votre serveur en ligne de commande.

*docker container exec -it some-nginx bash*

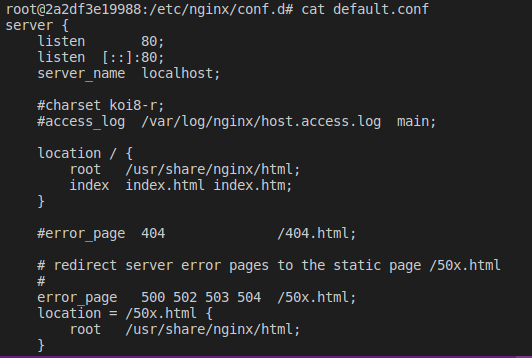
1. Déplacez-vous dans le répertoire de configuration et récupérez le fichier de configuration par défaut. Vous pouvez juste le copier dans le répertoire monté. Ouvrez le fichier de configuration et répondez aux questions suivantes :

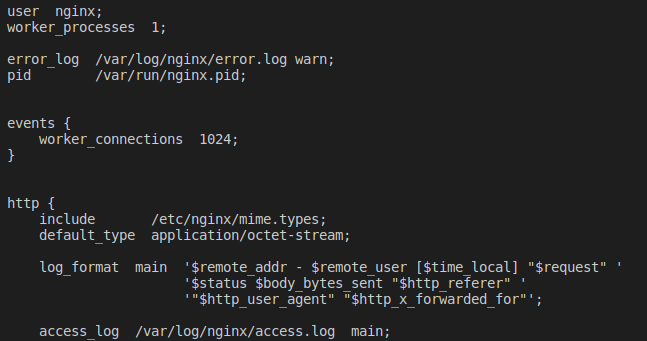
Consulter les sites de documentations de nginx pour vous aider :

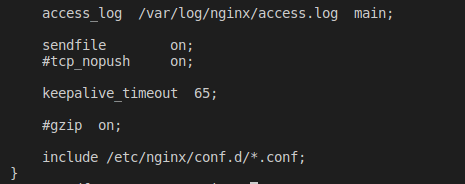
<https://nginx.org/en/docs/>

<https://docs.nginx.com/nginx/admin-guide/>

* 1. Quelle est l’utilité de la directive user ? Rép : Dans *etc*nginx/nginx.conf → indique quel utilisateur Nginx peut partir le processus du serveur web
  2. Quelle est l’utilité de la directive worker\_process ? Rép : *etc*/nginx/nginx.conf. Donne le nombre de processus
  3. À quel endroit se trouvent les logs et sont-ils gérés par docker ? Rép : /var/log/
  4. Pourquoi ne voit-on pas de contexte server dans le contexte http?







Rép : Car on inclut le répertoire conf.d/ qui a les configurations par défaut (et les serveurs)

* 1. Suite à la réponse de la question précédente, comment pouvons-nous implanter des hôtes virtuels avec cette version de nginx ? Un .conf à 8h58 10 novembre

Rép : Il faut « include vhosts/\*.conf; »

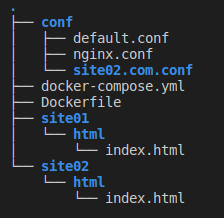
1. Récupérer le fichier /etc/nginx/conf.d/default.conf.

*cp default.conf /usr/share/nginx/html*

*cp nginx.conf /usr/share/nginx/html*

Étape 2 : Hôtes virtuels

1. À cette étape, avec ce que vous avez appris à l’étape précédente, vous allez refaire votre site avec des hôtes virtuels avec nginx. Votre structure de répertoire devrait ressembler à ça :

/SitesNginx/

|--docker-compose.yml

|--Dockerfile

|--conf/

| |--nginx.conf

| |--default.conf

| |--site02.com.conf #default.conf

|--site01/

| |--html/

| |--index.html

|--site02/

| |--html/

| |--index.html

* 1. Créer les fichiers nécessaires dans SitesNginx pour créer une image et un conteneur avec les spécifications suivantes (vos fichiers index.html doivent indiquer le nom du site et que le serveur est nginx) : Docker-compose

**Image à utiliser** : nginx:alpine.

**Dans l’image** : mettre à jour le serveur, copier les fichiers de configurations nécessaires et exposer les port 80 et 443. L’hôte par défaut est site01.com et on veut 2 processus worker pour le serveur.

**Répertoires à créer dans l’image** : /usr/share/nginx/html/site01.com et /usr/share/nginx/html/site02.com.

**Nom du conteneur** : nginx.

**Ports à exposer** : le port 80 au port 80 du conteneur.

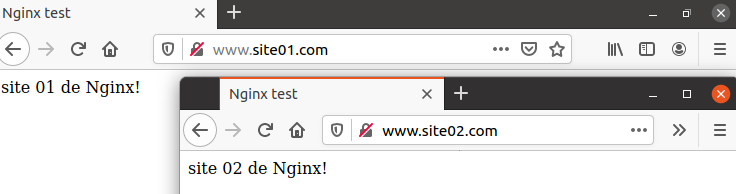
**Point de montage** : site01/html <-> /usr/share/nginx/html/site01.com et site02/html <-> /usr/share/nginx/html/site02.com. En lecture seulement.

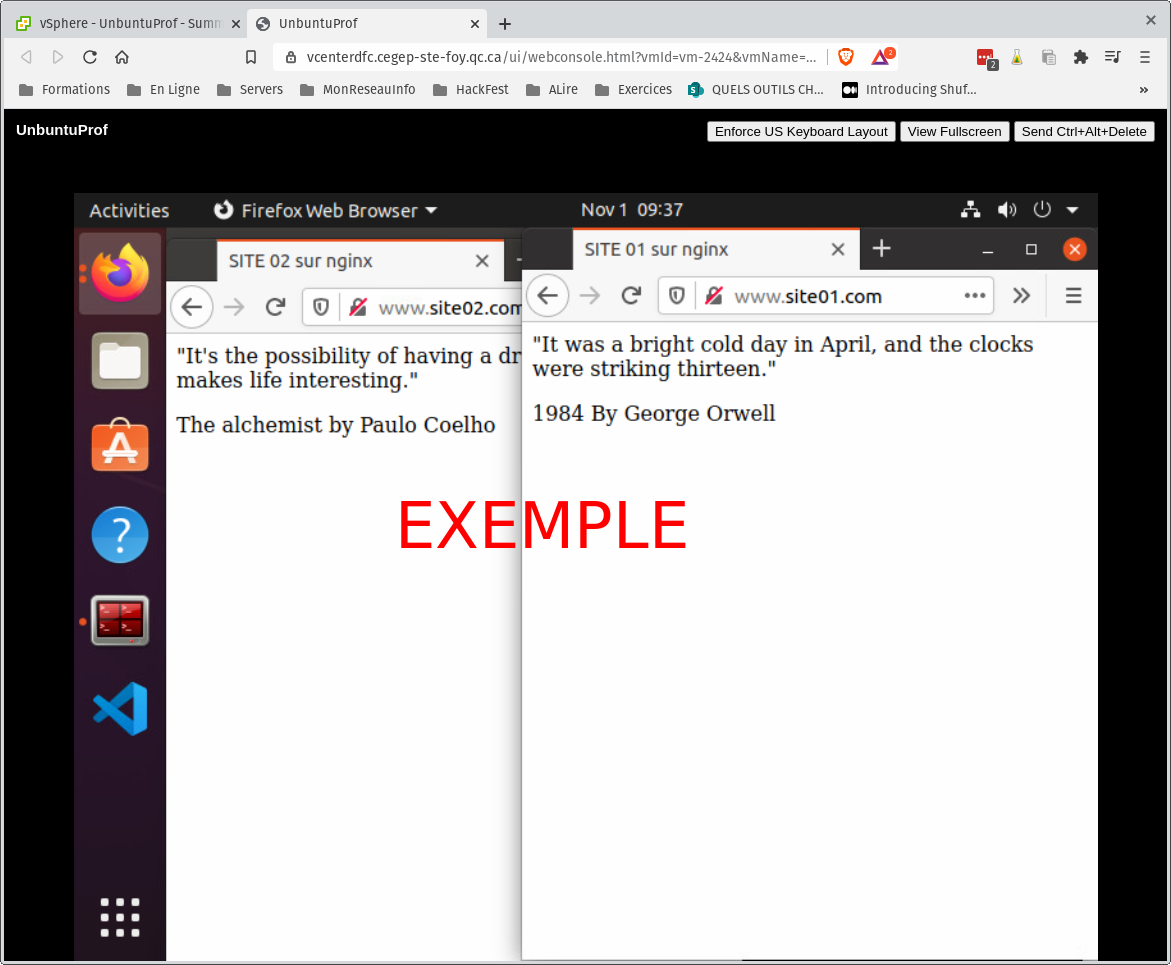
**Nom de domaine** : les sites doivent être accessibles sur votre VM avec site01.com, www.site01.com site02.com et [www.site02.com](http://www.site02.com/).

1. Lancer votre conteneur et vérifier que les deux sites fonctionnent.
2. Arrêter et effacer votre conteneur nginx.

Pour vérification

Remettre une capture d’écran de votre Linux avec Firefox ouvert sur les deux sites de votre serveur Web.





Compétences développées

|  |  |
| --- | --- |
| **00SJ** – Effectuer le déploiement de serveurs intranet. | **00SJ # 1** – Analyser le projet de déploiement.  **00SJ # 2** – Monter les serveurs intranet.  **00SJ # 3** – Installer les services intranet.  **00SJ # 5** – Participer à la mise en service des serveurs intranet. |

**Note** : les compétences sont développées en partie.

Références

<https://nginx.org/en/docs/>

<https://docs.nginx.com/nginx/admin-guide/>

<https://hub.docker.com/_/nginx>