



**NOUS ÉCLAIRONS.
VOUS BRILLEZ.**

420-W45-SF Installation de serveurs et de services

**Claude Roy
Serveur Web Nginx**



**FORMATION CONTINUE
ET SERVICES AUX ENTREPRISES**

Introduction

- nginx [engine x] est un serveur HTTP, un serveur proxy inverse, un serveur proxy de messagerie et un serveur proxy TCP/UDP générique,
- Un serveur Web occupé doit répondre à plusieurs milliers de requêtes simultanées.
- La plupart du temps nécessaire pour traiter chaque demande est passé à attendre que les données arrivent du réseau ou du disque.
- Le temps passé à traiter activement la demande est court par comparaison.

Nginx

- Pour gérer efficacement cette charge de travail, NGINX utilise un système basé sur les événements dans lequel seuls quelques processus de travail traitent plusieurs demandes simultanément.
- Lorsqu'une demande ou une réponse (un événement) est prête à être servie, un processus de travail termine rapidement le traitement avant de revenir pour gérer l'événement suivant.
- Surtout, NGINX vise à éviter le blocage sur les E/S réseau ou disque.

Processus maître (master)

- Deux processus: un maître et un travailleur.
- Le maître effectue des tâches administratives telles que l'ouverture de sockets, la lecture de la configuration et le maintien en cours d'exécution des autres processus NGINX.

Processus travailleur (worker)

- Les travailleurs font la plupart des gros travaux en recevant et en traitant les demandes.
- Certaines configurations utilisent des processus supplémentaires dédiés à la mise en cache.

Privilèges

- Comme dans Apache, le processus maître s'exécute en tant que root afin qu'il puisse ouvrir des sockets pour tous les ports inférieurs à 1024.
- Les autres processus s'exécutent en tant qu'utilisateur moins privilégié.

Nombres de processus

- Le nombre de processus de travail est configurable.
- Une bonne règle d'or consiste à exécuter autant de processus de travail que le système a de cœurs de processeur.

Nombres de processus

- Debian et Ubuntu configurent NGINX de cette façon par défaut s'il est installé à partir d'un paquet.
- FreeBSD et RHEL utilisent par défaut un seul processus de travail.

Fichier de configuration

- Le fonctionnement de nginx et de ses modules est déterminé dans le fichier de configuration.
- Fichier de configuration par défaut :
`nginx.conf`.
- Pour lancer nginx avec un fichier de configuration personnalisé :
`- nginx -c my_nginx.conf`

Démarrage, arrêt et rechargement de la configuration

- Nginx peut être contrôlé en invoquant l'exécutable avec le paramètre `-s` :

`- nginx -s signal`

signal

- *signal* peut être l'un des suivants:
 - **stop** - arrêt rapide
 - **quit** - arrêt normal
 - **reload** - recharger le fichier de configuration
 - **reopen** - rouvrir les fichiers journaux

Vérification du fichier de configuration

- Il est possible de vérifier le fichier de configuration avant de le recharger :
 - `nginx -t`

Informations nginx

- Sous Debian/Ubuntu

Nom du package	nginx
Path du daemon	/usr/sbin/nginx
Configuration root	/etc/nginx
Fichier de configuration	nginx.conf
Utilisateur par défaut	www-data

Structure de nginx.conf

- nginx se compose de modules contrôlés par des directives spécifiées dans le fichier de configuration.
- Les directives sont divisées en directives simples et directives de bloc.

Directives simples

- Une directive simple se compose du nom et des paramètres séparés par des espaces et se termine par un point-virgule (;).

Directives de bloc

- Une directive de bloc a la même structure qu'une directive simple, mais au lieu du point-virgule, elle se termine par un ensemble d'instructions supplémentaires entourées d'accolades ({ et }).

Contexte

- Si une directive de bloc peut avoir d'autres directives entre accolades, elle est appelée un contexte (exemples: events, http, server et location).

Exemple

```
events {}  
http {  
    server {  
        server_name www.monsite.com;  
        root /var/www/admin.com;  
    }  
}
```

- Voici une configuration minimale.
- Le contexte le plus externe (appelé `main`) est implicite et configure la fonctionnalité de base.

Events

- Les contextes `events` et `http` vivent dans `main`.
- `events` est un contexte obligatoire qui configure la gestion des connexions.
- Puisqu'il est vide dans cet exemple, les valeurs par défaut sont implicites.

Valeurs par défauts de `events`

- Exécutez un processus de travail (utilisez le compte d'utilisateur non privilégié).
- Écoutez sur le port 80 s'il est démarré en tant que root ou sur le port 8000 sinon.
- Écrire les journaux dans `/var/log/nginx` (choisi au moment de la compilation).

http

- Contiens toutes les directives relatives aux services de proxy Web et HTTP.

server

- Définis des hôtes virtuels.
- Imbriqués dans `http`.
- Plusieurs contextes `server` dans `http` configureraient plusieurs hôtes virtuels.

Site Web de référence

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