

# GUIDE D'INSTALLATION D'UN SERVEUR NGINX SOUS LINUX OPENSUSE

Auteur : Leonce Wadje Kamgo

## Introduction

Ce guide explique comment installer et configurer un serveur **Nginx** sous **Linux OpenSUSE** pour héberger un site web sur un serveur nommé **kingpin**. Chaque étape comprend les commandes nécessaires et des captures d'écran pour vérifier le bon déroulement de l'installation.

## Prérequis

Avant de commencer, assurons-Nous d'avoir :

- Un serveur fonctionnant sous OpenSUSE.
- Un accès avec des privilèges **root** ou un utilisateur avec les droits **sudo**.
- Une connexion Internet active.

## Étapes d'installation

### 1. Renommage du serveur

Avant d'installer Nginx, nous allons renommer le serveur en **kingpin**.

Commande :

```
sudo hostnamectl set-hostname kingpin
```

```
leonce@localhost:~> sudo hostnamectl set-hostname kingpin
leonce@localhost:~> hostnamectl
  Static hostname: kingpin
        Icon name: computer-vm
        Chassis: vm
        Machine ID: f7c6827a9e7b4144b974d0294d670fb7
        Boot ID: 0df532bc76ce4c149f4d82c0d48bfaf4
  Virtualization: vmware
Operating System: openSUSE Leap 15.6
   CPE OS Name: cpe:/o:opensuse:leap:15.6
     Kernel: Linux 6.4.0-150600.23.38-default
  Architecture: x86_64
   Hardware Vendor: VMware, Inc.
   Hardware Model: VMware Virtual Platform
Firmware Version: 6.00
   Firmware Date: Thu 2020-11-12
   Firmware Age: 4y 4month 6d
leonce@localhost:~> .....
```

## 2. Installation de Nginx

Nginx n'est pas installé par défaut sur OpenSUSE. Nous devons l'ajouter et l'activer.

Commandes :

`sudo zypper install nginx`

```
leonce@localhost:~> sudo zypper refresh
1
Récupération des métadonnées du dépôt 'Update repository of openSUSE Backports' .....[fait]
Construction du cache du dépôt 'Update repository of openSUSE Backports' .....[fait]
Récupération des métadonnées du dépôt 'Update repository with updates from SUSE Linux Enterprise 15' .....[fait]
Construction du cache du dépôt 'Update repository with updates from SUSE Linux Enterprise 15' .....[fait]
Le dépôt 'Dépôt principal de mise à jour' est à jour.
Le dépôt 'Dépôt de mises à jour (Non-OSS)' est à jour.
Le dépôt 'openSUSE-Leap-15.6-1' est à jour.
Le dépôt 'Dépôt Non-OSS' est à jour.
Le dépôt 'Open H.264 Codec (openSUSE Leap)' est à jour.
Le dépôt 'Dépôt principal' est à jour.
Tous les dépôts ont été rafraichis.
leonce@localhost:~> sudo zypper install nginx
2
Chargement des données du dépôt...
Lecture des paquets installés...
'nginx' est déjà installé.
Pas de candidat à la mise à jour pour 'nginx-1.21.5-150600.10.3.1.x86_64'. La version la plus récente est déjà installée. 3
Résolution des dépendances des paquets...
Rien à faire.
leonce@localhost:~> |
```

## 3. Activation et démarrage de Nginx

Après l'installation, nous devons activer et démarrer le service.

Commandes :

`sudo systemctl start nginx`

```
leonce@localhost:~> sudo systemctl enable nginx
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service -> /usr/lib/systemd/system/nginx.service.
leonce@localhost:~> sudo systemctl start nginx
leonce@localhost:~> sudo systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: disabled)
   Active: active (running) since Thu 2025-03-20 00:42:41 EDT; 16s ago
     Process: 14507 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
    Main PID: 14509 (nginx)
       Tasks: 2 (limit: 2250)
          CPU: 242ms
      CGroup: /system.slice/nginx.service
              └─14509 "nginx: master process /usr/sbin/nginx -g daemon off;"
                  └─14510 "nginx: worker process"

mars 20 00:42:41 kingpin systemd[1]: Starting The nginx HTTP and reverse proxy server...
mars 20 00:42:41 kingpin nginx[14507]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
mars 20 00:42:41 kingpin nginx[14507]: nginx: configuration file /etc/nginx/nginx.conf test is successful
mars 20 00:42:41 kingpin systemd[1]: Started The nginx HTTP and reverse proxy server.
leonce@localhost:~> |
```

## 4. Configuration du pare-feu

Nous devons autoriser le trafic HTTP et HTTPS pour permettre l'accès au serveur.

Commandes :

- ✓ `sudo firewall-cmd --add-service=https --permanent`
- ✓ `sudo firewall-cmd --reload`

```
leonce@localhost:~>
leonce@localhost:~> sudo firewall-cmd --add-service=https --permanent
[sudo] Mot de passe de root :
success
leonce@localhost:~> sudo firewall-cmd --reload
success
leonce@localhost:~> sudo firewall-cmd --list-services
dhcpv6-client https ssh
leonce@localhost:~>
```

## 5. Création d'un répertoire pour le site web

Nous allons créer un répertoire pour notre site et y placer une page de test.

Commandes :

- ✓ `sudo mkdir -p /srv/www/kingpin`
- ✓ `sudo echo "<h1>Bienvenue sur Kingpin</h1>" | sudo tee /srv/www/kingpin/index.html`

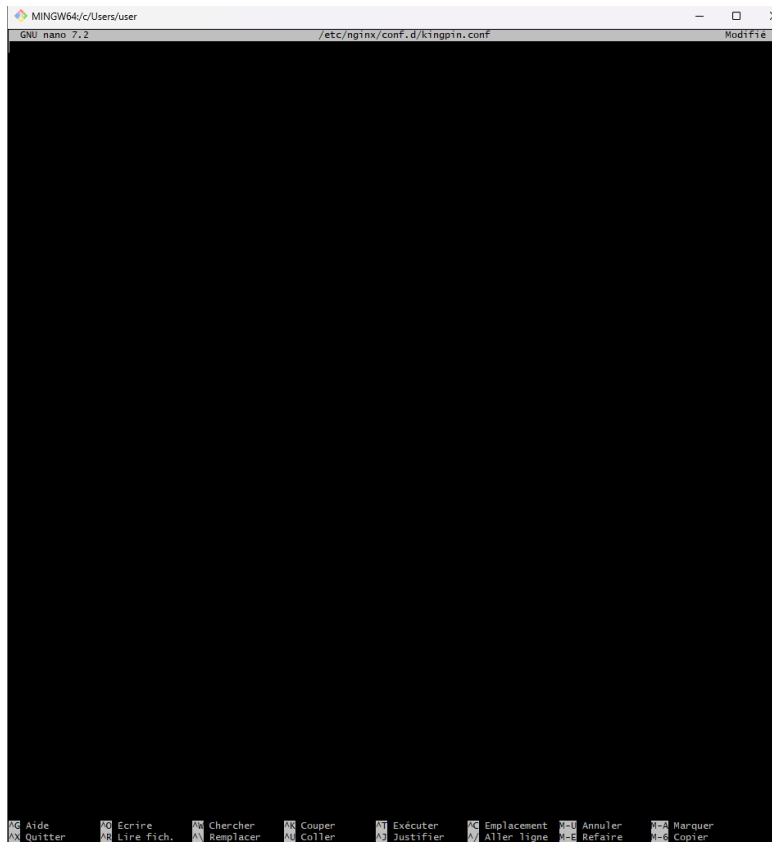
```
leonce@localhost:~> sudo mkdir -p /srv/www/kingpin
leonce@localhost:~> sudo echo "<h1>Bienvenue sur Kingpin</h1>" | sudo tee /srv/www/kingpin/index.html
<h1>Bienvenue sur Kingpin</h1>
leonce@localhost:~>
```

## 6. Configuration de Nginx

Nous allons configurer un hôte virtuel pour notre site **kingpin**.

Commandes :

`sudo nano /etc/nginx/conf.d/kingpin.conf`



Ajoutons le contenu suivant :

```
server {  
    listen 80;  
    server_name kingpin;  
    root /srv/www/kingpin;  
    index index.html;  
    location / {  
        try_files $uri $uri/ =404;  
    }  
}
```

Enregistrez et quittez (*CTRL + X, Y, Entrée*).

```
leonce@localhost:~>  
leonce@localhost:~>  
leonce@localhost:~> sudo firewall-cmd --add-service=https --permanent  
[sudo] Mot de passe de root :  
success  
leonce@localhost:~> sudo firewall-cmd --reload  
success  
leonce@localhost:~> sudo firewall-cmd --list-services  
dhcpv6-client https ssh  
leonce@localhost:~> sudo mkdir -p /srv/www/kingpin  
leonce@localhost:~> sudo echo "<h1>Bienvenue sur Kingpin</h1>" | sudo tee /srv/www/kingpin/index.html  
<h1>Bienvenue sur Kingpin</h1>  
leonce@localhost:~>  
leonce@localhost:~>  
leonce@localhost:~> sudo nano /etc/nginx/conf.d/kingpin.conf  
[sudo] Mot de passe de root :  
leonce@localhost:~> |
```

Vérification et redémarrage :

`sudo nginx -t`

```
leonce@localhost:~> sudo nginx -t
[sudo] Mot de passe de root :
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
leonce@localhost:~> |
```

## 7. Ajout du nom de domaine localement

Pour tester le site localement, nous devons modifier le fichier **hosts**.

Commande :

`echo "127.0.0.1 kingpin" | sudo tee -a /etc/hosts`

```
leonce@localhost:~>
leonce@localhost:~>
leonce@localhost:~> echo "127.0.0.1 kingpin" | sudo tee -a /etc/hosts
127.0.0.1 kingpin
leonce@localhost:~> |
```

## 8. Preuve de fonctionnement

Nous allons tester notre site web via un navigateur ou la ligne de commande.

Commande :

`curl http://kingpin`

```
leonce@localhost:~> curl http://kingpin
<h1>Bienvenue sur Kingpin</h1>
leonce@localhost:~> |

leonce@localhost:~> curl -I http://kingpin
HTTP/1.1 200 OK
Server: nginx/1.21.5
Date: Thu, 20 Mar 2025 05:37:31 GMT
Content-Type: text/html
Content-Length: 31
Last-Modified: Thu, 20 Mar 2025 04:52:14 GMT
Connection: keep-alive
ETag: "67db9efe-1f"
Accept-Ranges: bytes

leonce@localhost:~> ^C
leonce@localhost:~> |
```

## Conclusion

Ce guide nous a permis d'installer et de configurer un serveur Nginx sur OpenSUSE pour héberger un site web sous le nom **kingpin**.