

Architecture

Prérequis

un seul serveur où sera installé le serveur de la base de donnée et les applications web :

- Serveur de base de donnée Mysql 8
- Docker (plateforme de conteneurs)
- Nginx (reverse proxy)
- apps (sous forme de conteneur docker)
- iadmin (sous forme de conteneur docker)
- ibuy (sous forme de conteneur docker)
- sdm (sous forme de conteneur docker)
- ilogistics (sous forme de conteneur docker)
- qr (sous forme de conteneur docker)
- public (sous forme de conteneur docker)

Spécifications du serveur

- Hardware
 - 48GO RAM minimum (6GO min pour chaque application pour un bon fonctionnement)
 - Une bonne Allocation CPU
 - 500GB minimum d'espace disque dur (pour accueillir le nombre important des fichier téléchargés depuis les applications)
- Software
 - Une installation fraîche de Ubuntu Server 20.04 (avec openssh-server)
 - <https://releases.ubuntu.com/focal/ubuntu-20.04.6-live-server-amd64.iso>
- Le serveur doit être connecté à internet (pour l'installation des applications et les prochaines MAJ des applications)

Installation

Installation de la base de donnée,docker et serveur nginx

Après l'installation Ubuntu Server 20.04 et la connexion avec le compte de l'utilisateur personnel, téléchargez et lancez le script d'installation a l'aide de ces commandes

```
cd ~  
git clone https://github.com/azdadanass/orangeinit.git  
cd orangeinit  
chmod +x *.sh  
./install.sh
```

le script de l'installation install.sh contient les détails de l'installation sous forme de commentaires

Après l'installation il faut nécessairement redémarrer la machine

```
sudo reboot
```

Installation de portainer

portainer est l'outil qui va permettre la gestion et la visualisation des logs des applications

pour installer portainer, lancez le script suivant avec le compte de l'utilisateur personnel

```
~/orangeinit/deploy-portainer.sh
```

Après l'exécution de la commande, vous pouvez accéder à l'interface depuis ce lien

```
https://ip_de_la_machine:9443
```

Cliquez sur Paramètres avancés, puis sur Continuer vers le site...



Votre connexion n'est pas privée

Des individus malveillants tentent peut-être de subtiliser vos informations personnelles sur le site **10.11.6.182** (mots de passe, messages ou numéros de carte de crédit, par exemple). [En savoir plus](#)

NET:ERR_CERT_AUTHORITY_INVALID

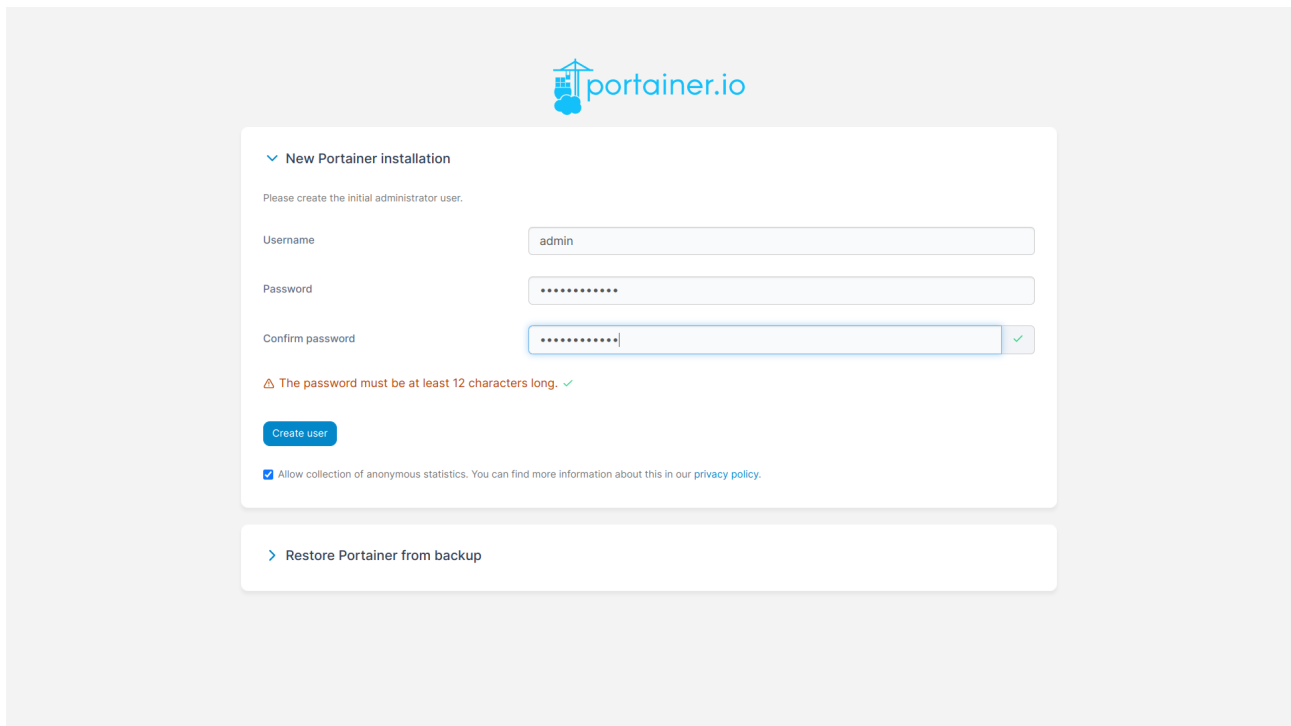


Pour bénéficier du niveau de sécurité le plus élevé de Chrome, [activez la protection renforcée](#)

Paramètres avancés

Revenir en lieu sûr

Le formulaire de création du compte admin vous sera proposé, créez le mot de passe admin puis cliquez sur create user



▼ New Portainer installation

Please create the initial administrator user.

Username

admin

Password

••••••••

Confirm password

••••••••

✓

⚠ The password must be at least 12 characters long. ✓


Create user

☒ Allow collection of anonymous statistics. You can find more information about this in our [privacy policy](#).

> Restore Portainer from backup


Dans le menu a gauche sélectionnez Home

Upgrade to Business Edition

 **portainer.io**
COMMUNITY EDITION

<<

Home

Environment:  None selected

Settings

Users

Environments

Registries





Authentication logs


Notifications

Settings

Environment Wizard

Quick Setup


   admin 

 Environment Wizard

Welcome to Portainer


We have connected your local environment of Docker to Portainer.

Get started below with your local portainer or connect more container environments.




Get Started

Proceed using the local environment which Portainer is running in

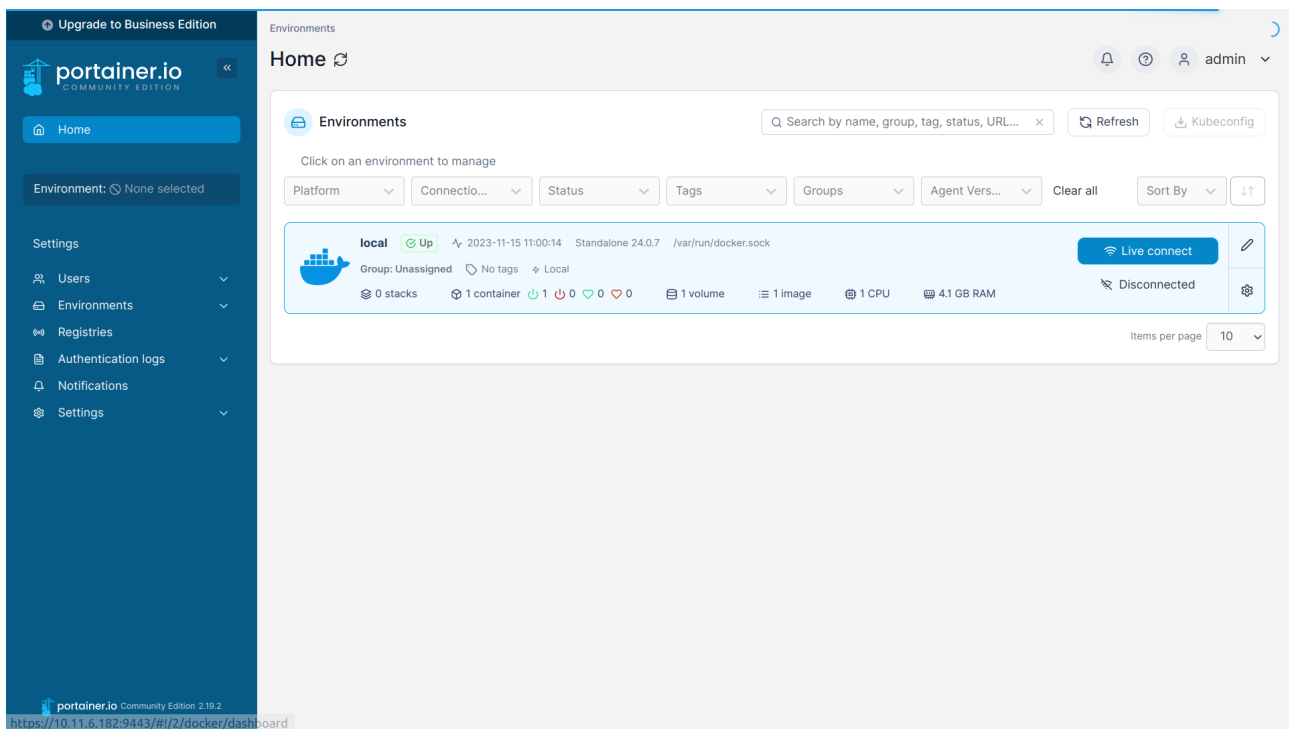


Add Environments

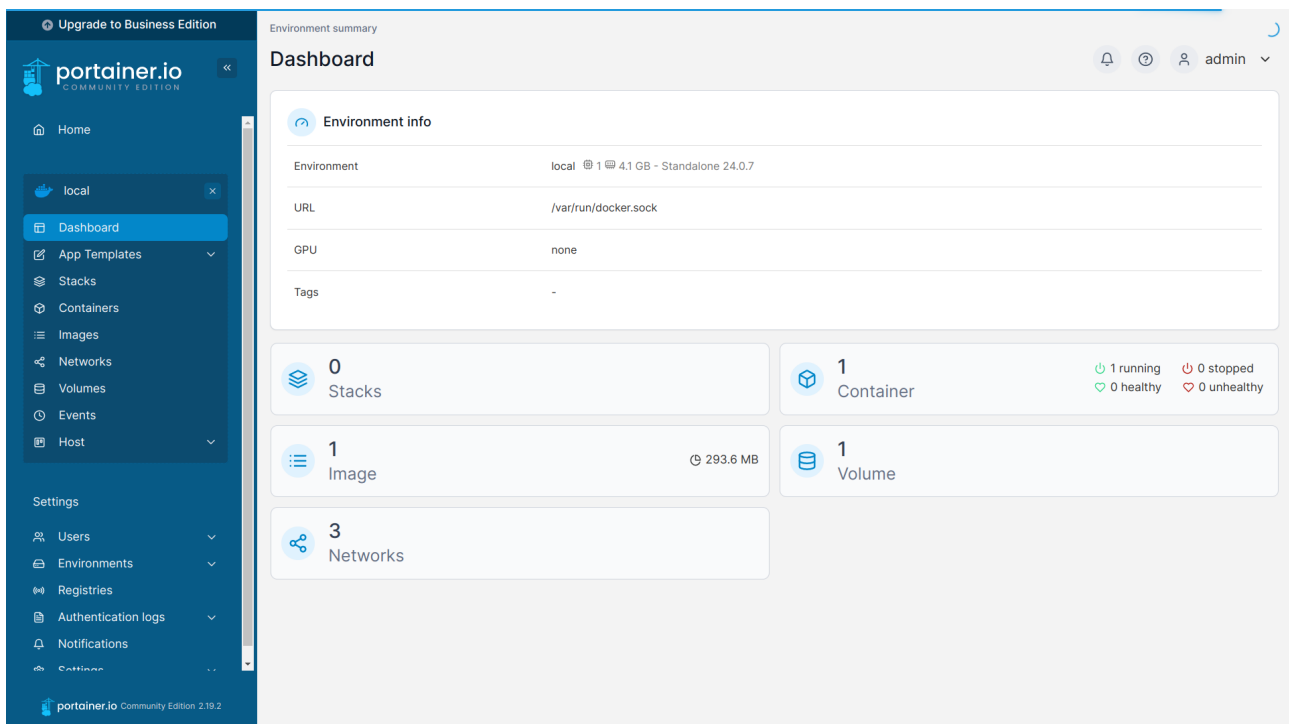
Connect to other environments

 **portainer.io** Community Edition 2.19.2

Cliquez sur l'environnement local



Cliquer sur Container



Dans cette page, vous trouverez la liste des applications installées (pour l'instant seulement portainer est installé)

Upgrade to Business Edition

portainer.io

COMMUNITY EDITION

Home

local

Dashboard

App Templates

Stacks

Containers

Images

Networks

Volumes

Events

Host

Settings

Users

Environments

Registries

Authentication logs

Notifications

Settings

portainer.io Community Edition 2.19.2

Containers

Container list

Containers

Search...

Start

Stop

Kill

Restart

Pause

Resume

Remove

Add container

Name	State	Filter	Quick Actions	Stack	Image	Created	IP Address	Published Ports	Ownership
portainer	running		<div></div>	-	portainer/portainer-ce:latest	2023-11-15 10:57:48	172.17.0.2	9443:9443 8000:8000	administrators

Items per page 10

Mémo­ri­sez cette page, ça va nous per­mettre d'accé­der aux logs des appli­cations que nous allons installer par la suite

Installation des applications (apps,iadmin ,sdm,ilogistics,public,qr,ibuy)

Pour installer ou faire la mise à jour des applications il suffit de lancer les commandes suivantes avec le compte de l'utilisateur personnel (chaque commande correspond a une application)

```
~/orangeinit/deploy-apps.sh
```

```
~/orangeinit/deploy-iadmin.sh
```

```
~/orangeinit/deploy-sdm.sh
```

```
~/orangeinit/deploy-ilogistics.sh
```

```
~/orangeinit/deploy-public.sh
```

```
~/orangeinit/deploy-qr.sh
```

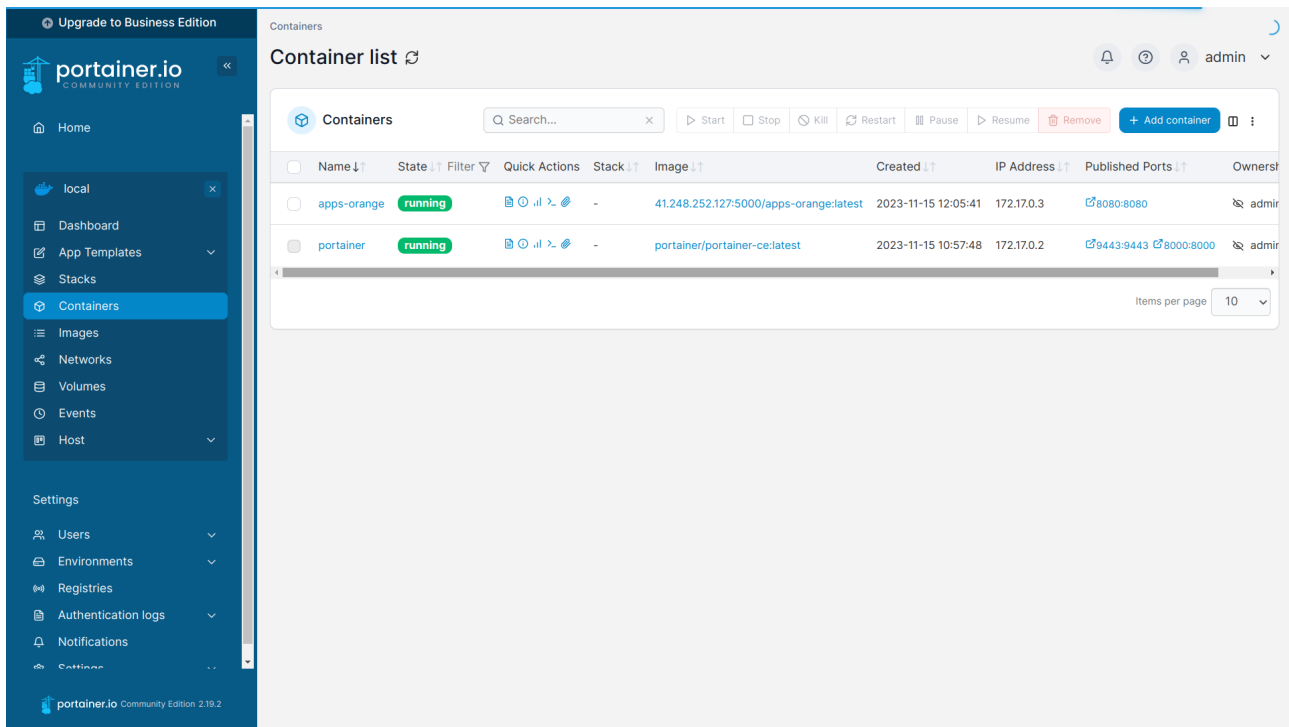
```
~/orangeinit/deploy-ibuy.sh
```

NB : pour la mise à jour d'une application , il faut coordonner avec le prestataire avant de lancer la commande de maj

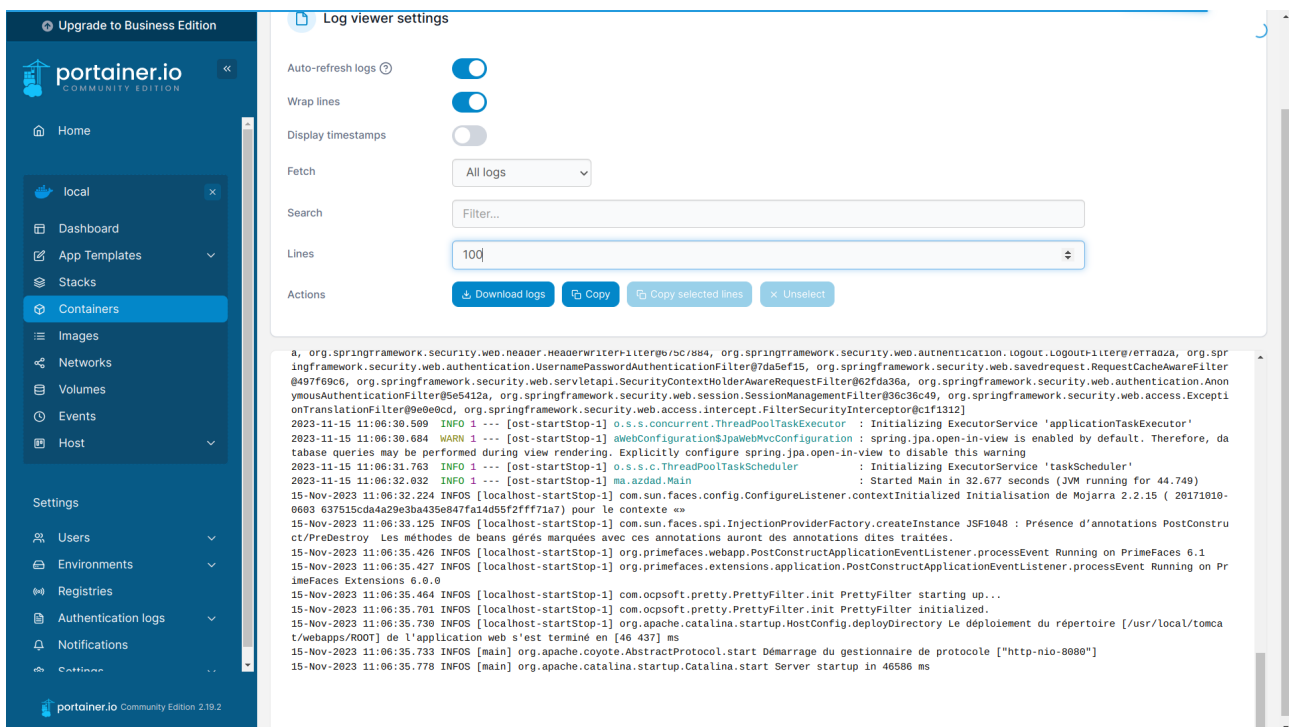
Exemple du lancement de apps

```
azdad@orange:~$ ~/orangeinit/deploy-apps.sh
Error response from daemon: No such container: apps-orange
Error response from daemon: No such container: apps-orange
Error response from daemon: No such image: 41.248.252.127:5000/apps-orange:latest
latest: Pulling from apps-orange
43f89b94cd7d: Downloading [=====>] 6.744MB/30.44MB
43f89b94cd7d: Pull complete
7f82deb1f5c2: Pull complete
3e4406cb8375: Pull complete
d2268b5b2c51: Pull complete
740734a399a8: Pull complete
b58bed7efd39: Pull complete
381dcddb7602: Pull complete
e6d62ef24dc2: Pull complete
a5eb02d47296: Pull complete
a4439d8751c4: Pull complete
1b7994b1fb07: Pull complete
814affce6096: Pull complete
dd6613b29f0c: Pull complete
4f4fb700ef54: Pull complete
c80df3b7b58: Pull complete
a5c438c7603e: Pull complete
be93f69a3d88: Pull complete
9694bc29acf3: Pull complete
3b02a572e5ca: Pull complete
Digest: sha256:c7e3729aad58ee5af6fc74999444690cc1ea85c5589ecc01891e177d7ddd030f
Status: Downloaded newer image for 41.248.252.127:5000/apps-orange:latest
41.248.252.127:5000/apps-orange:latest
28c9bfb4d83793a5ed9ab04881be6bdc8f7f4838b59821ba043e381eab8183e8
azdad@orange:~$ |
```

Après l'exécution de la commande `~/orangeinit/deploy-apps.sh`, vous allez voir apparaître un nouveau container dans l'interface portainer



vous pouvez voir le log de apps-orange en cliquant sur l'icône de log dans Quick Actions, le log vous permettra de voir si le lancement de apps s'est bien déroulé



Si tout s'est bien déroulé, vous pouvez tester l'application apps-orange depuis ce lien :

http://ip_de_la_machine:8080

orange™



Please Enter Your Information

Username

Password

Login

[← I forgot my password](#)

pour les autres applications :

Apps	<u>http://ip_de_la_machine:8080</u>
iadmin	<u>http://ip_de_la_machine:8081</u>
sdm	<u>http://ip_de_la_machine:8082</u>
ilogistics	<u>http://ip_de_la_machine:8083</u>
public	<u>http://ip_de_la_machine:8084</u>
qr	<u>http://ip_de_la_machine:8085</u>
ibuy	<u>http://ip_de_la_machine:8086</u>

NB : ces liens seront utilisés dans la configuration du reverse proxy nginx. Dans la partie qui viendra après, les ports [8080-8086] seront désactivés pour forcer seulement l'accès https (443)

Configuration pour accès HTTPS

Pour le moment, nos applications sont accessibles depuis les port [8080-8086] en http

pour passer en https, vous devez suivre ces étapes

Prenons l'exemple de iadmin (supposons que le nom de domaine qui a été attribué à iadmin par l'équipe de DNS est `iadmin.example.com`)

Nous devons commencer par créer le CSR

```
sudo mkdir -p /etc/nginx/ssl/  
sudo openssl req -new -newkey rsa:2048 -nodes -keyout  
/etc/nginx/ssl/iadmin.key -out /etc/nginx/ssl/iadmin.csr
```

Vous devez remplir le formulaire comme suit (Common Name étant le champs le plus important, vous devez entrez le nom de domaine de iadmin fourni par l'équipe DNS, dans notre exemple c'est `iadmin.example.com`)

```
Country Name : MA  
State : Casablanca  
Locality Name : Casablanca  
company : orange  
Organizational Unit Name : delivery  
Common Name : iadmin.example.com  
Email address : vous pouvez laisser ce champs vide  
A challenge password : vous pouvez laisser ce champs vide  
An optional company name : vous pouvez laisser ce champs vide
```

Deux fichiers seront générés dans le dossier `/etc/nginx/ssl` ,qui sont `iadmin.key` et `iadmin.csr`

Vous devez fournir le fichier `iadmin.csr` à l'équipe responsable de la génération des certificat https qui vont générer le certificat (`iadmin.crt`)

Il suffit maintenant de placer le fichier crt reçu dans `/etc/nginx/ssl/iadmin.crt` (la nomination du fichier est importante)

Vous devez faire cette action pour les autres applications :

- apps
- sdm
- ilogistics
- qr
- public
- ibuy

Configuration du reverse proxy nginx

Pour configurer le reverse proxy il suffit de lancer la commande

```
sudo nano /etc/nginx/sites-enabled/default
```

puis corriger le domaine example.com par le domaine fourni par l'équipe responsable des DNS

```
client_max_body_size 100M;

#####
# apps
#####

server {
    listen 80;
    server_name apps.example.com;
    return 301 https://$host$request_uri;
}

server {
    listen 443 ssl;
    server_name apps.example.com;
    ssl_certificate /etc/nginx/ssl/apps.crt;
    ssl_certificate_key /etc/nginx/ssl/apps.key;
    ssl_session_cache builtin:1000 shared:SSL:10m;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers HIGH:!aNULL:!eNULL:!EXPORT:!CAMELLIA:!DES:!MD5:!PSK:!RC4;
    ssl_prefer_server_ciphers on;
    access_log /var/log/nginx/apps.access.log;
    location / {
        proxy_set_header X-Forwarded-Host $host;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $host;
        proxy_pass http://localhost:8080;
    }
}

#####
# iadmin
#####

server {
    listen 80;
    server_name iadmin.example.com;
    return 301 https://$host$request_uri;
}

server {
    listen 443 ssl;
    server_name iadmin.example.com;
    ssl_certificate /etc/nginx/ssl/iadmin.crt;
    ssl_certificate_key /etc/nginx/ssl/iadmin.key;
    ssl_session_cache builtin:1000 shared:SSL:10m;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers HIGH:!aNULL:!eNULL:!EXPORT:!CAMELLIA:!DES:!MD5:!PSK:!RC4;
    ssl_prefer_server_ciphers on;
    access_log /var/log/nginx/iadmin.access.log;
    location / {
        proxy_set_header X-Forwarded-Host $host;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $host;
    }
}
```

Après la sauvegarde du fichier (Ctr x + Yes), vous devez redémarrer nginx avec cette commande :

```
sudo service nginx restart
```

Après l'exécution de cette dernière commande, les applications devront être accessibles depuis les liens corrects en mode https

```
https://apps.example.com
```

```
https://iadmin.example.com
```

```
https://sdm.example.com
```

```
https://ilogistics.example.com
```

```
https://public.example.com
```

```
https://qr.example.com
```

```
https://ibuy.example.com
```

Activation et configuration du par-feu

Pour configurer le par-feu, il suffit de lancer la commande suivante

```
~/orangeinit/ufw.sh
```

Après l'exécution de cette dernière commande, seulement les ports 80 (http) et 443 (https) seront accessibles