# Architecture

### L’architecture est détaillée dans le document architecture.pdf

### En gros elle est constituée de deux serveurs : Serveur principale et serveur de backup

Pour le serveur de backup, il faut juste installer ubuntu server (comme expliquer dans architecture.pdf ) avec openssh-server

par la suite les étapes d’installation du serveur principale

# **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

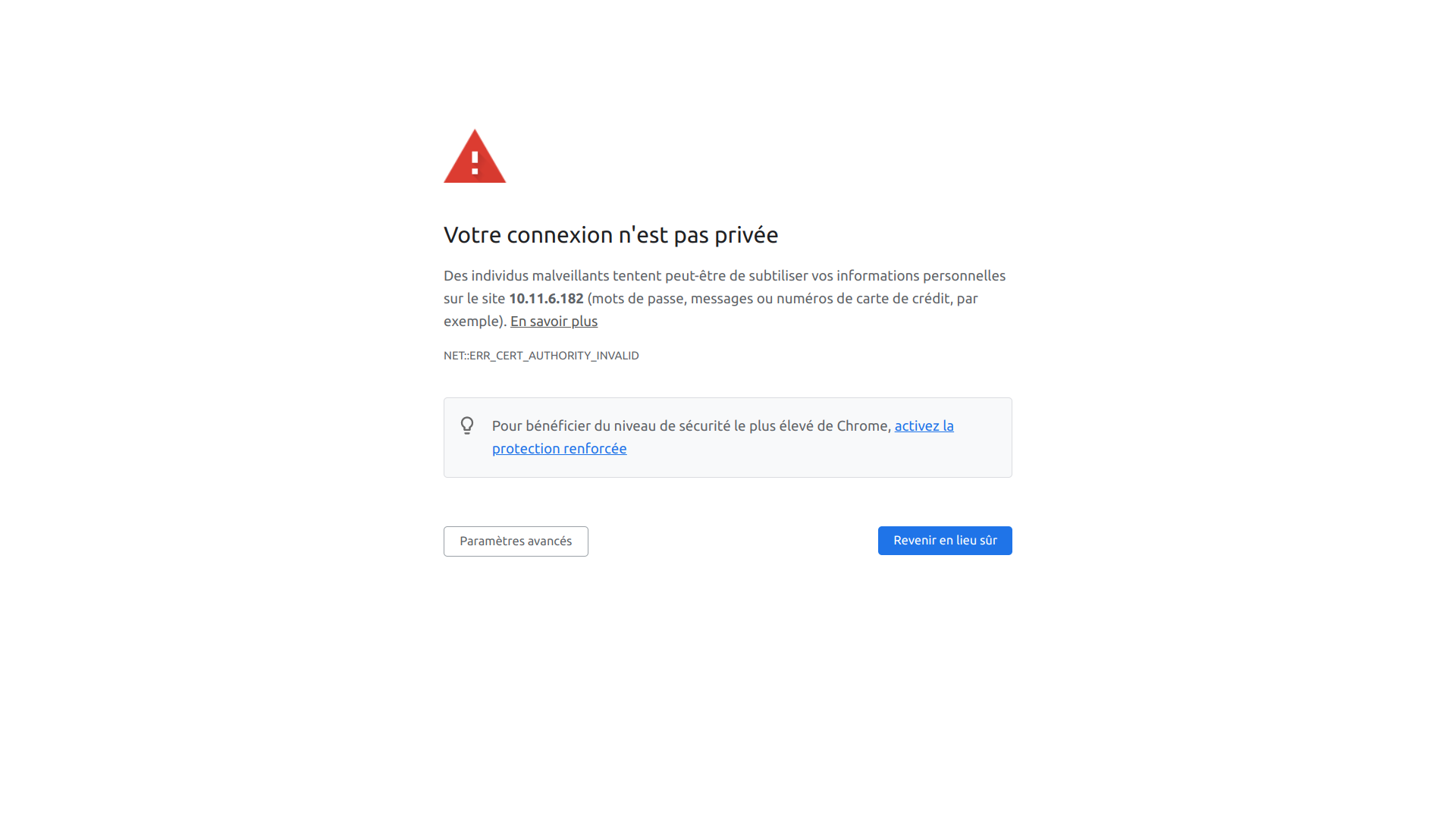
pour installer portainer, lancez le script suivant avec le compte de l’utilisateur personnel

~/orangeinit/deploy-portainer.sh

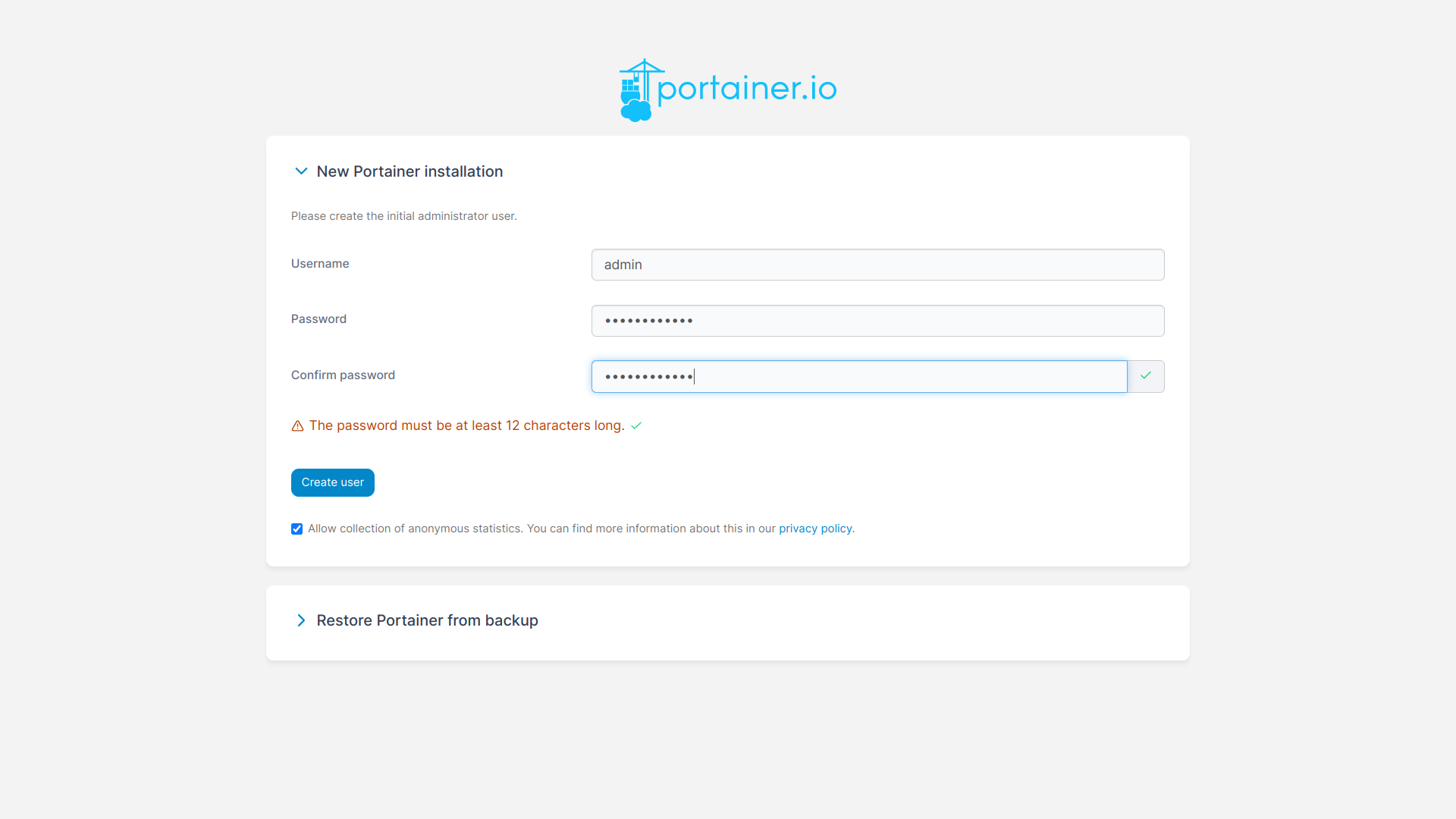
Après l exécution de la commande, vous pouvez accéder à l’interface depuis ce lien

https://ip\_de\_la\_machine:9443

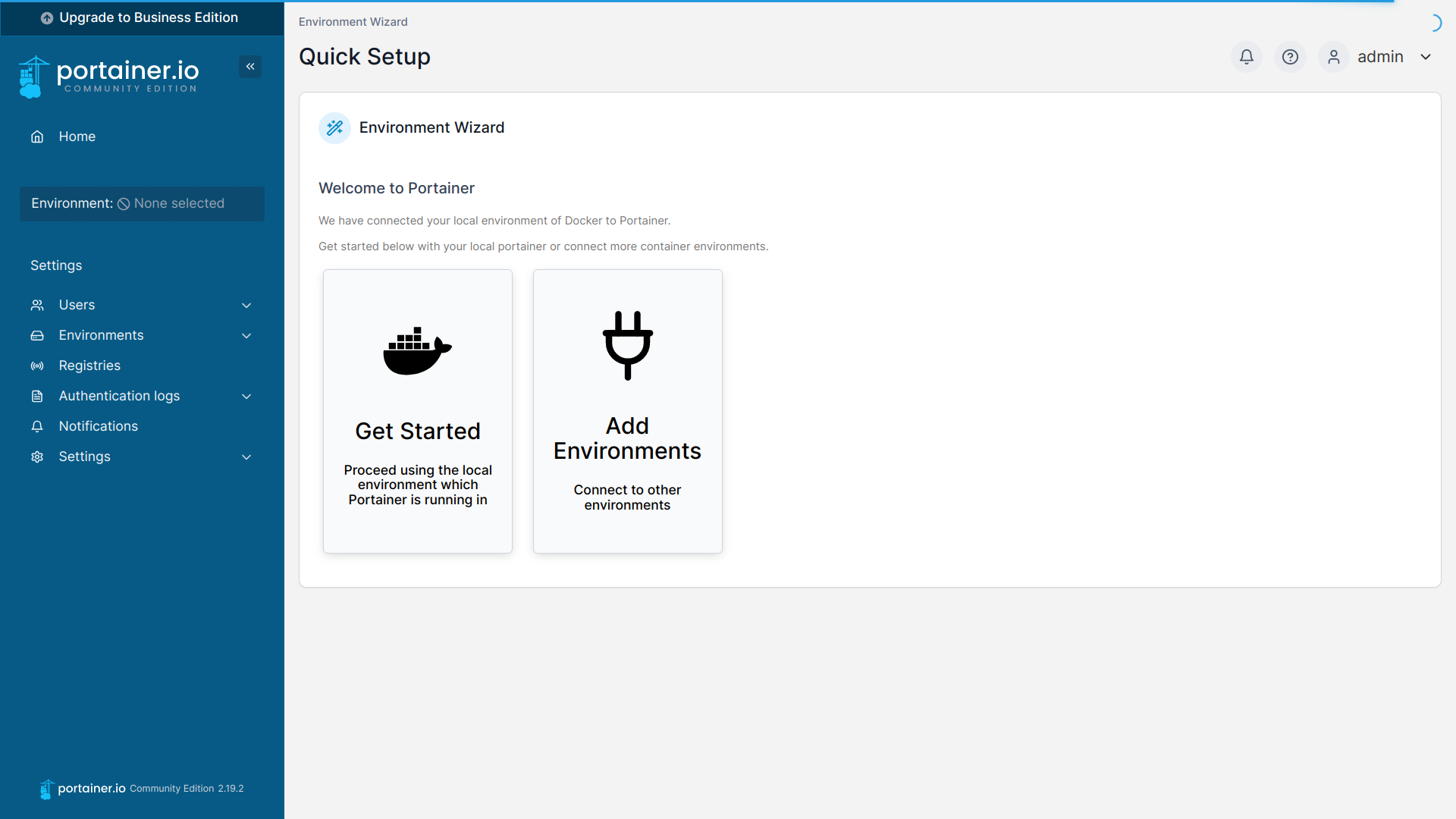
Cliquez sur Paramètres avancés, puis sur Continuer vers le site…



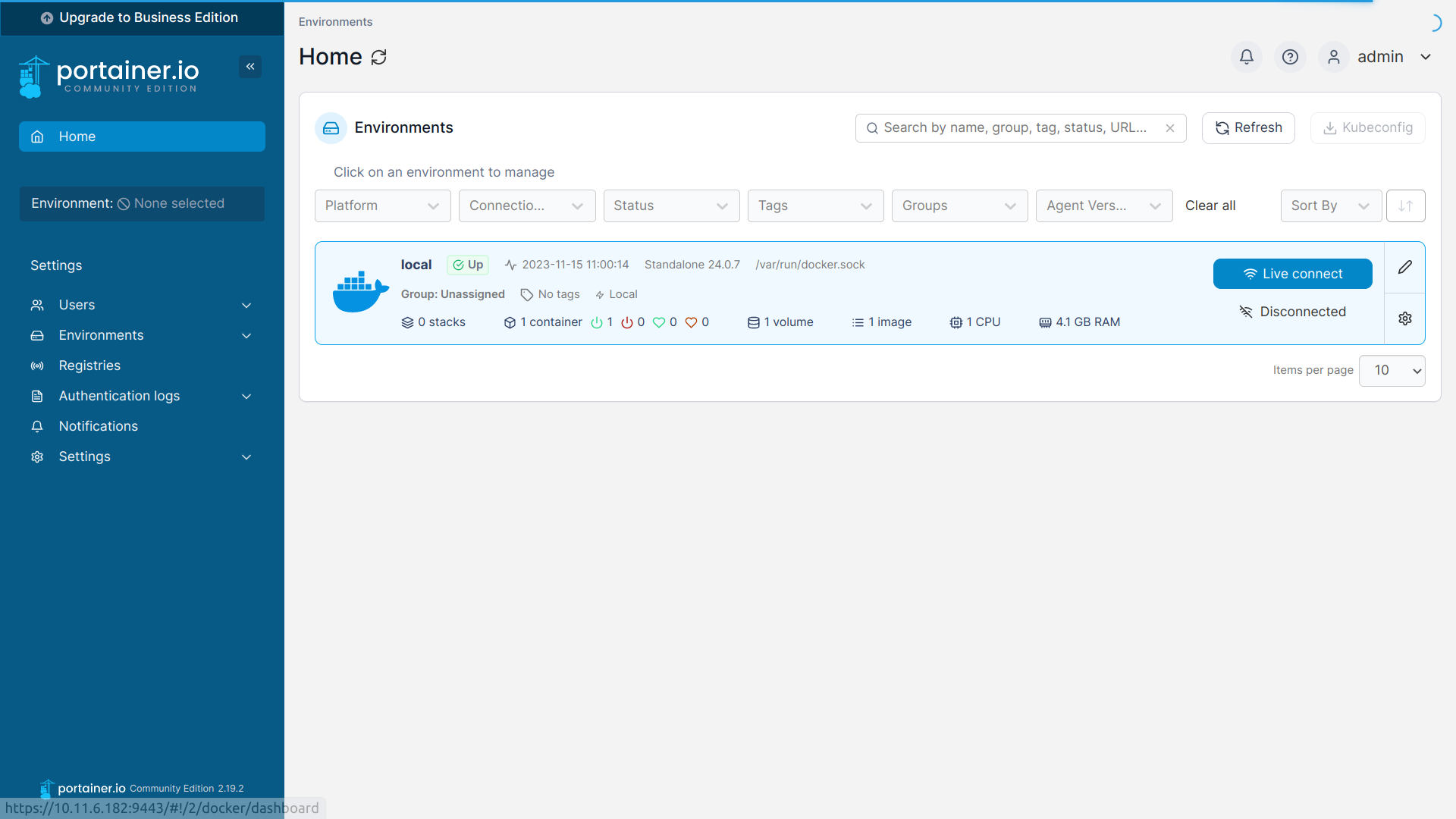
Le formulaire de création du compte admin vous sera proposé,créez le mot de passe admin puis cliquez sur create user



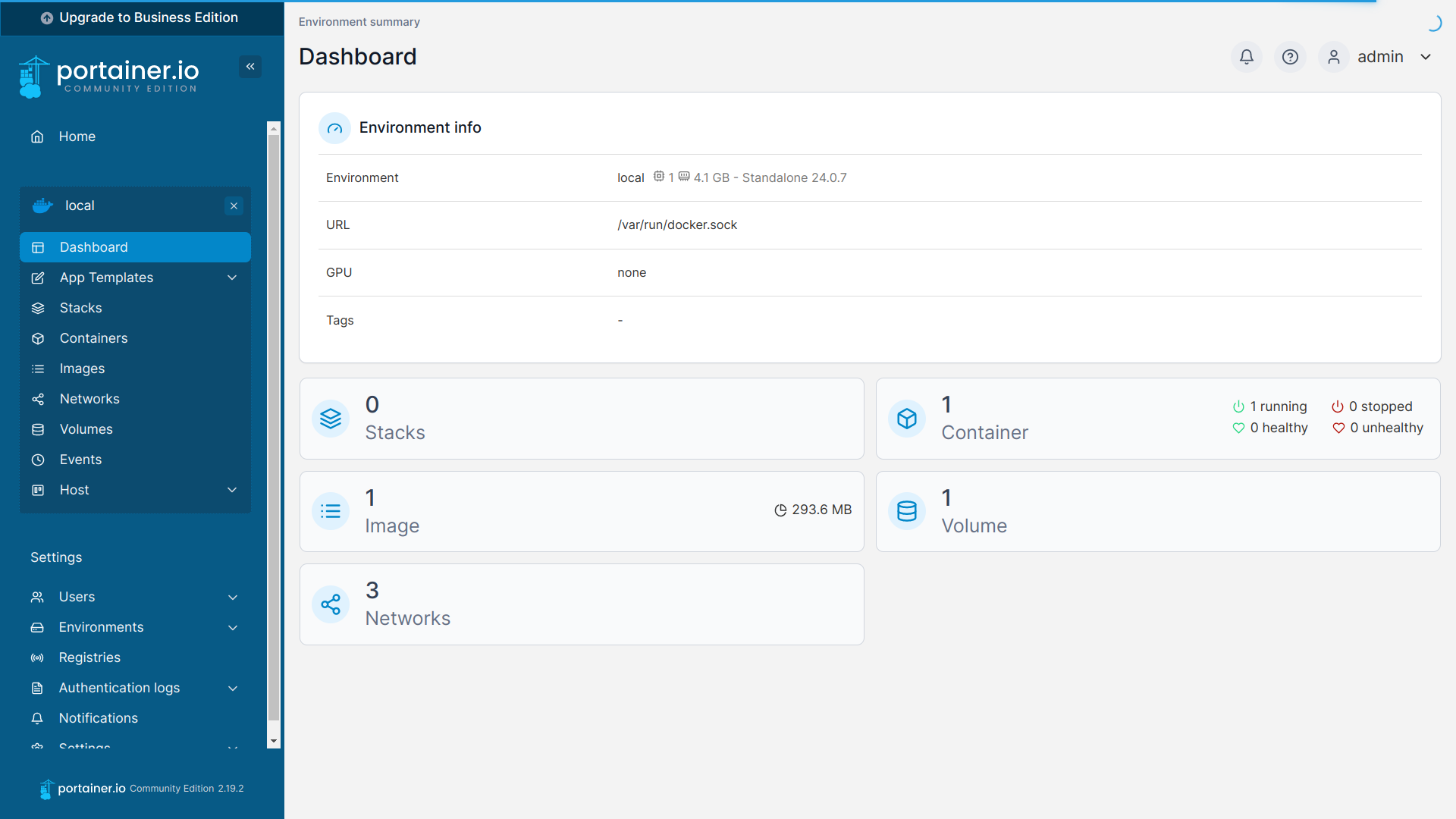
Dans le menu a gauche sélectionnez Home



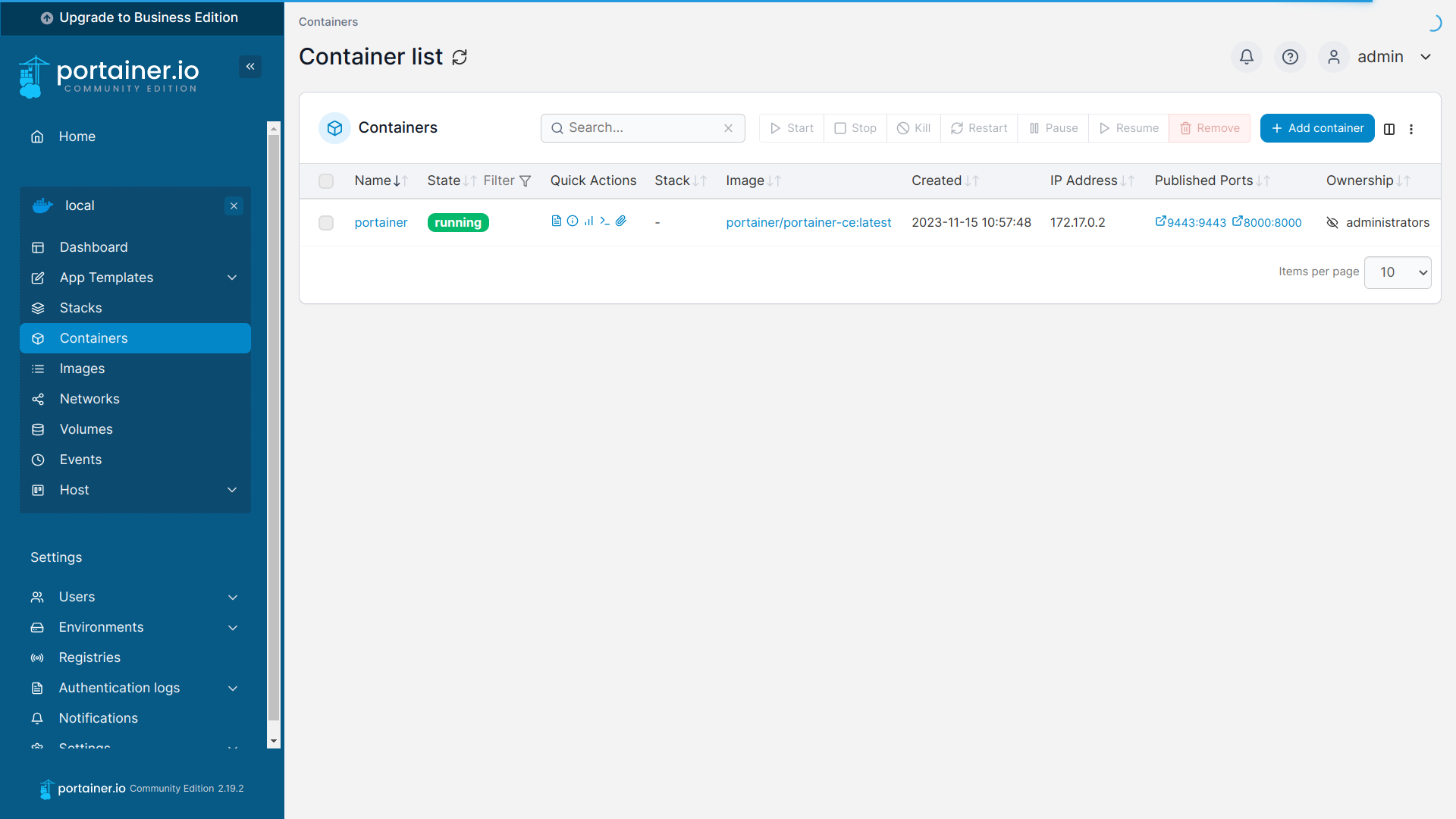
Cliquez sur l environnement local



Cliquer sur Container



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



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

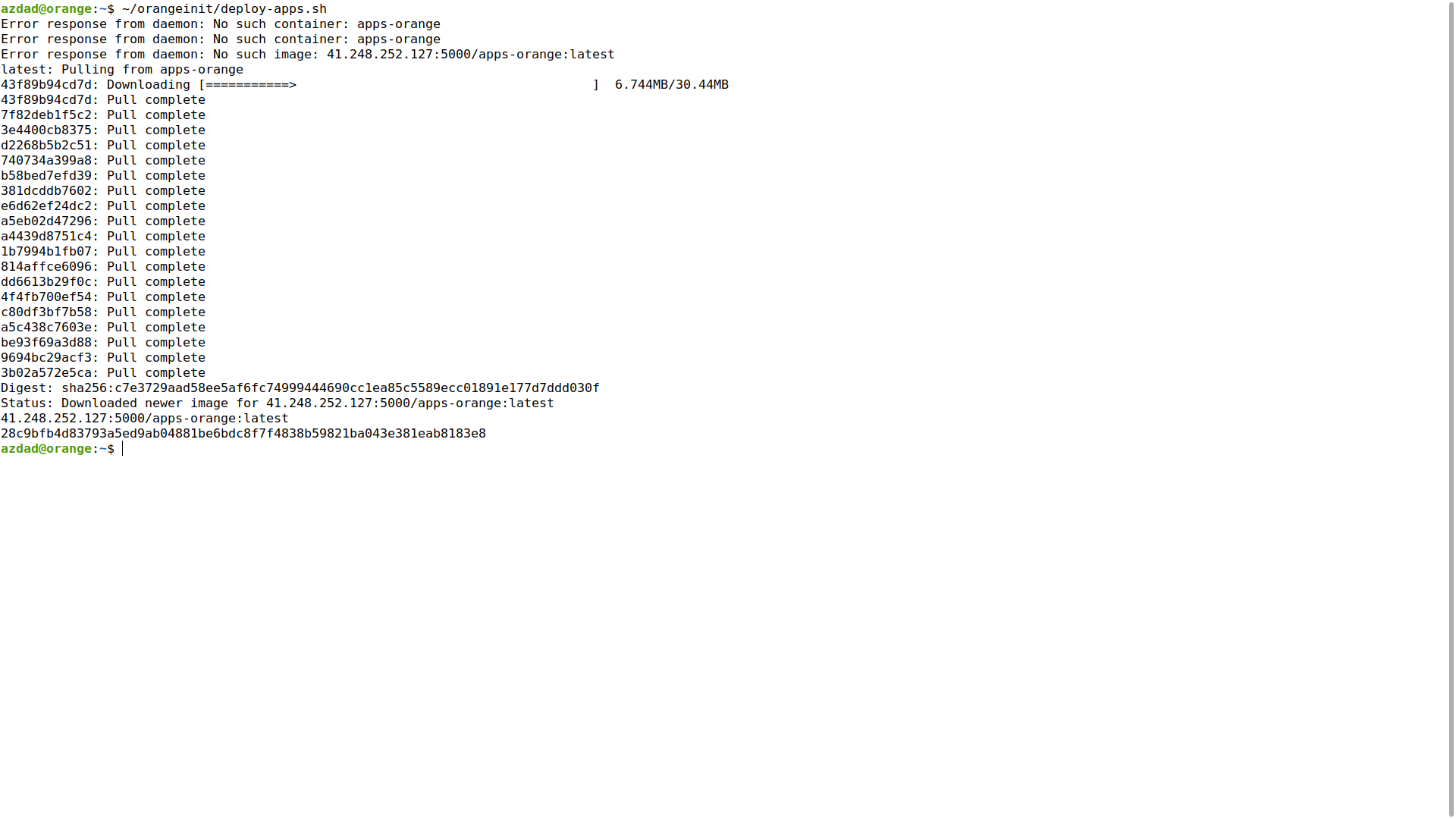
~/orangeinit/deploy-public.sh

~/orangeinit/deploy-qr.sh

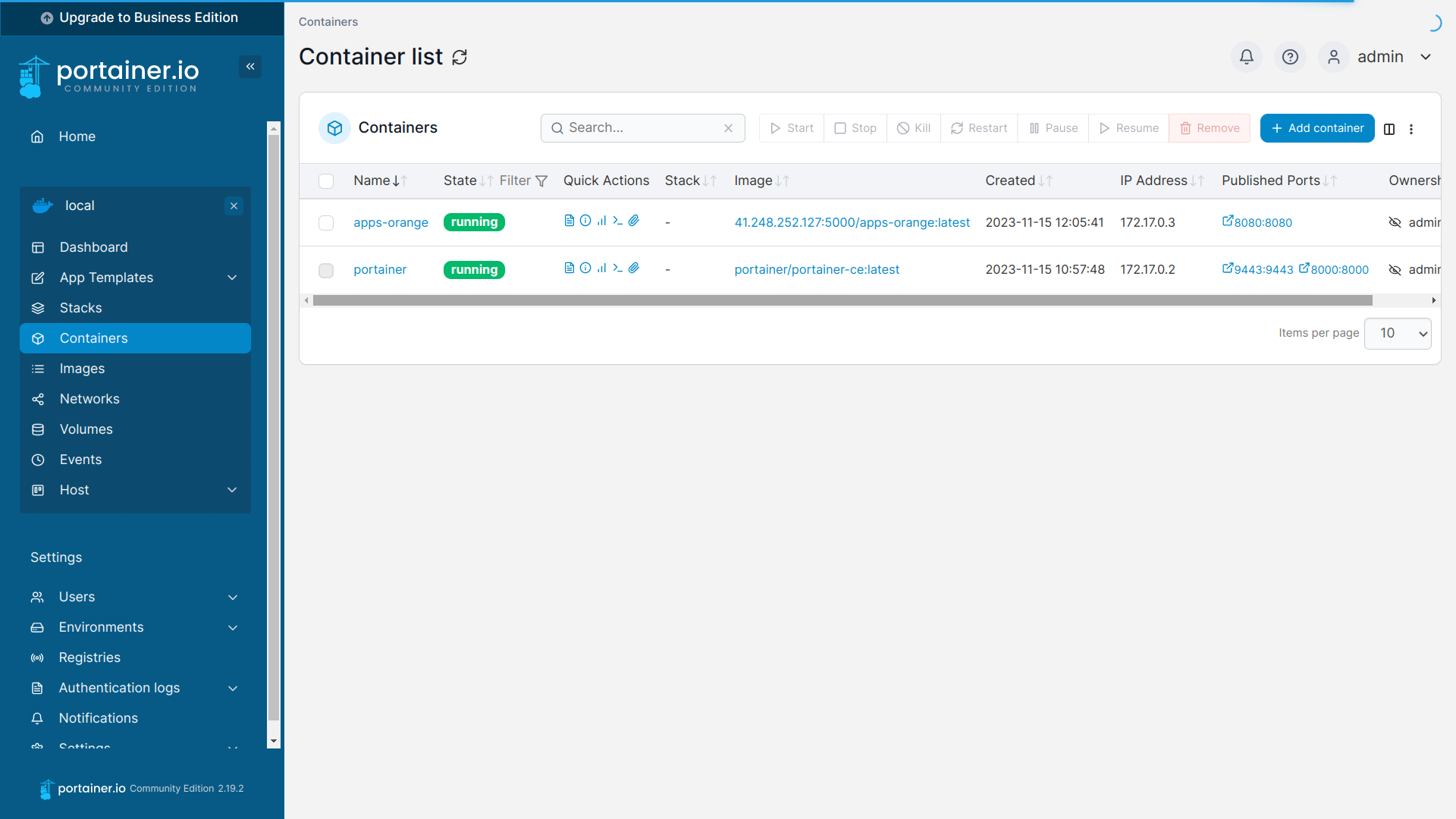
~/orangeinit/deploy-ibuy.sh

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

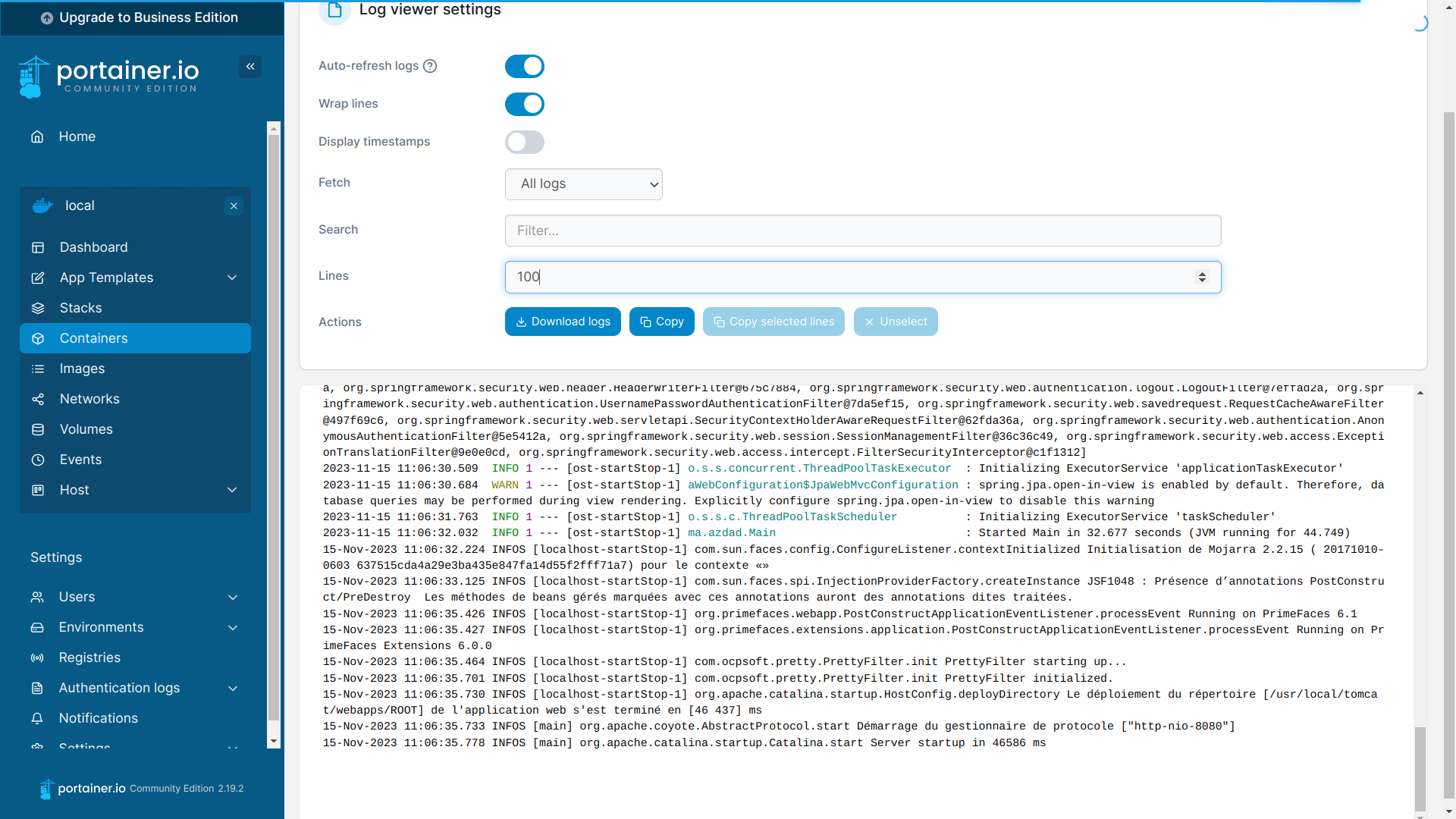
Exemple du lancement de apps



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

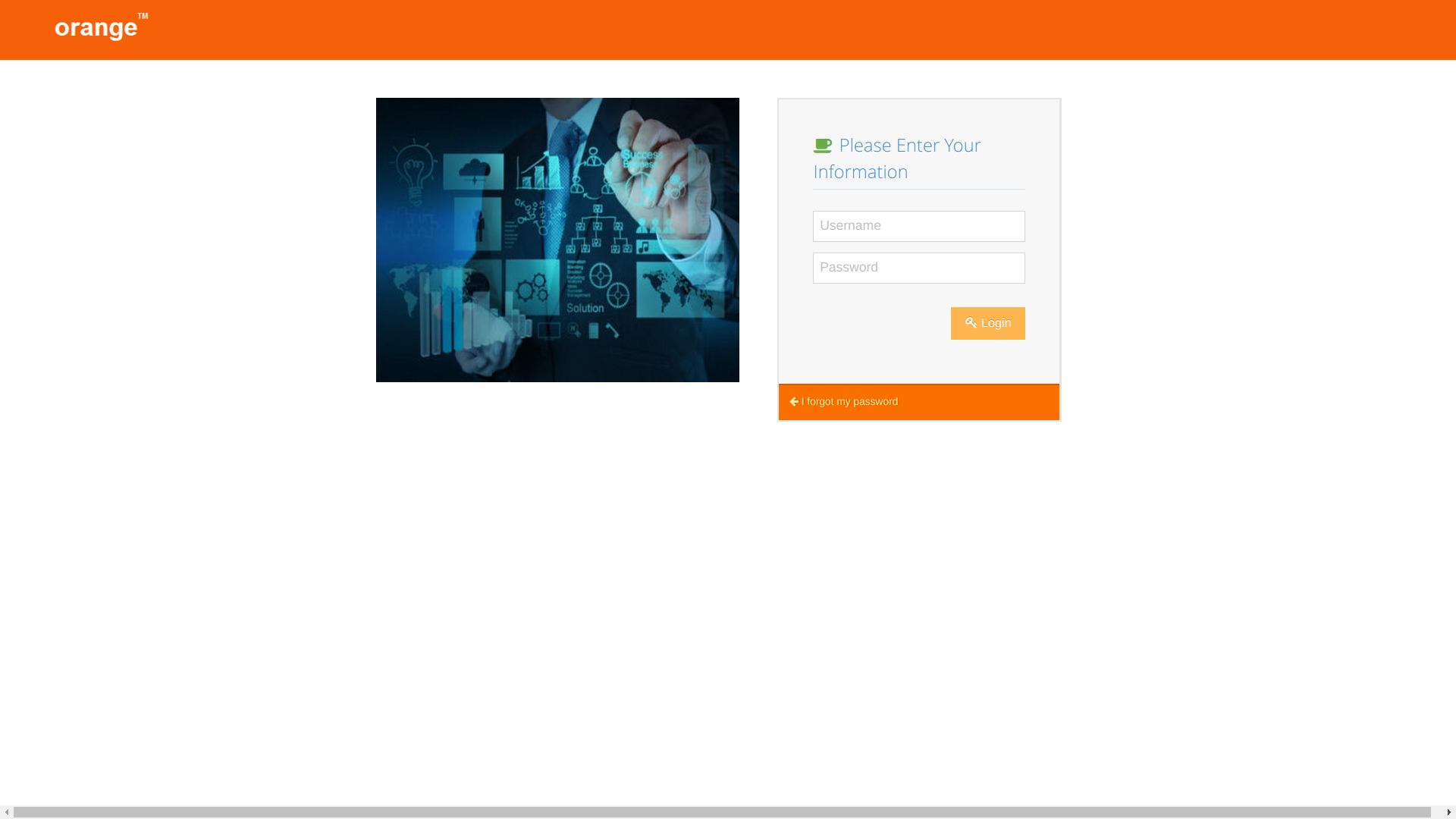


vous pouvez voir le log de apps-orange en cliquant sur l’icône de log dans Quick Actions, le log vous permettra de voir si le lancement de apps s’est bien déroulé



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

http://ip\_de\_la\_machine:8080



pour les autres applications :

|  |  |
| --- | --- |
| Apps | [http://ip\_de\_la\_machine:8080](http://ip_de_la_machine:8080/) |
| iadmin | [http://ip\_de\_la\_machine:808](http://ip_de_la_machine:8080/)1 |
| sdm | [http://ip\_de\_la\_machine:808](http://ip_de_la_machine:8080/)2 |
| ilogistics | [http://ip\_de\_la\_machine:808](http://ip_de_la_machine:8080/)3 |
| public | [http://ip\_de\_la\_machine:808](http://ip_de_la_machine:8080/)4 |
| qr | [http://ip\_de\_la\_machine:808](http://ip_de_la_machine:8080/)5 |
| ibuy | [http://ip\_de\_la\_machine:808](http://ip_de_la_machine:8080/)6 |

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) ou http (80)

# **Configuration du reverse proxy nginx**

#### Configuration pour accès HTTP

Pour configurer le serveur pour accès HTTP, lancez cette commande

~/orangeinit/config.sh

Dans l’invite de commande , entrez les informations suivantes :

* l’ip locale du serveur principale (exemple : 192.168.1.100)
* le DNS qui sera utilisé pour les applications (exemple telodigital.orange.com) ce parametre sera fourni par l’équipe responsable du DNS
* dans protocol, entrez http

Normalement les applications devront être accessible depuis les liens corrects (exemple si on suppose que le dns était telodigital.orange.com)

* http://apps.telodigital.orange.com
* http://iadmin.telodigital.orange.com
* http://sdm.telodigital.orange.com
* http://ilogistics.telodigital.orange.com
* http://ibuy.orange.telodigital.com
* http://qr.telodigital.orange.com
* http://public.telodigital.orange.com

#### **Configuration pour accès HTTPS**

1. Pour configurer le serveur pour accès HTTPS, il faut d’abord générer le certificat (fichier crt) pour chaque application
2. Prenons l’exemple de iadmin
3. Nous devons commencer par créer le CSR
4. sudo mkdir -p /etc/nginx/ssl/
5. sudo openssl req -new -newkey rsa:2048 -nodes -keyout /etc/nginx/ssl/iadmin.key -out /etc/nginx/ssl/iadmin.csr
6. 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.telodigtal.orange.com)
7. Country Name : MA
8. State : Casablanca
9. Locality Name : Casablanca
10. company : orange
11. Organizational Unit Name : delivery
12. Common Name : iadmin.telodigital.orange.com
13. Email address : vous pouvez laisser ce champs vide
14. A challenge password : vous pouvez laisser ce champs vide
15. An optional company name : vous pouvez laisser ce champs vide
16. Deux fichiers seront générés dans le dossier /etc/nginx/ssl ,qui sont iadmin.key et iadmin.csr
17. Vous devez fournir le fichier iadmin.csr à l’équipe responsable de la génération des certificat https qui vont générer le certificat (iadmin.crt)
18. Il suffit maintenant de placer le fichier crt reçu dans /etc/nginx/ssl/iadmin.crt (la nomination du fichier est importante)
19. Vous devez faire cette action pour les autres applications :

* apps
* sdm
* ilogistics
* qr
* public
* ibuy

Maintenant que les fichiers crt sont bien placé il suffit de lancer cette commande

1. ~/orangeinit/config.sh
2. Dans l’invite de commande , entrez les informations suivantes :

* l’ip locale du serveur principale (exemple : **192.168.1.100**)
* le DNS qui sera utilisé pour les applications (exemple **telodigital.orange.com**) ce paramètre sera fourni par l’équipe responsable du DNS
* dans protocol, entrez **https**

les applications devront être accessible en HTTPS

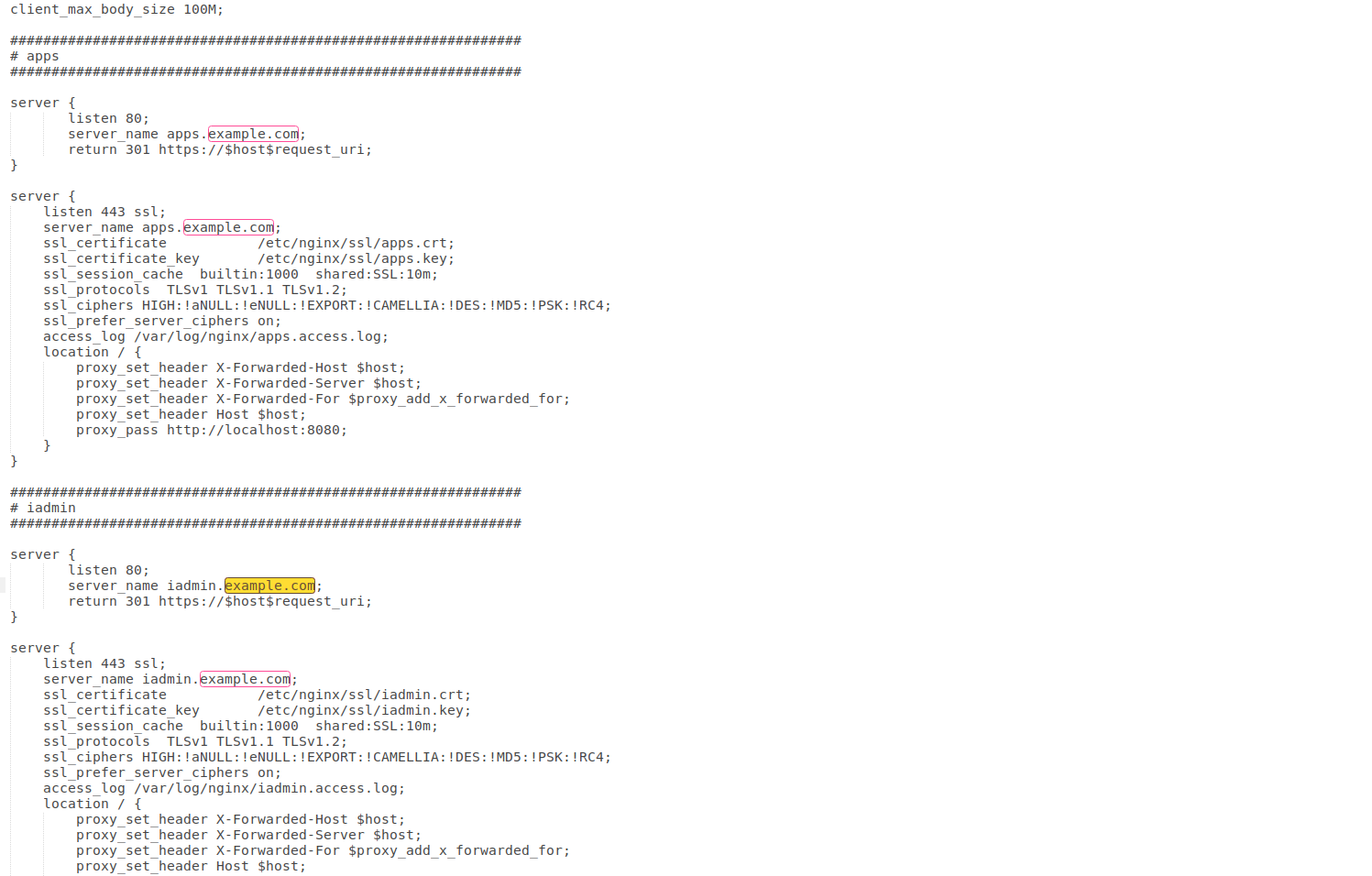
* https://apps.telodigital.orange.com
* https://iadmin.telodigital.orange.com
* https://sdm.telodigital.orange.com
* https://ilogistics.telodigital.orange.com
* https://ibuy.orange.telodigital.com
* https://qr.telodigital.orange.com
* https://public.telodigital.orange.com

### 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



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, seuls les ports 80 (http) et 443 (https) seront accessible