

Installation et utilisation de Portainer

Version de Debian : 12

1 – Installation de portainer.....	1
2 – Configuration de portainer.....	2
3 – Créer des containers avec Portainer	3
4 – Onglet Stacks	6
5 – Onglet images	7
6 – Onglet Networks.....	9
7 – Onglet Volumes	10

Portainer permet la supervision des containers Docker via une interface web. La gestion devient donc centralisée et simplifiée.

Il est nécessaire que Docker soit installé sur votre machine avant de faire l'installation de Portainer, pour installer Docker merci de suivre ce lien : <https://docs.docker.com/engine/install/debian/>

1 – Installation de portainer

En premier lieu, nous allons devoir créer un volume Docker afin d'héberger les données de portainer

```
sudo docker volume create portainer_data
```

```
root@SRVA-DOCKER:~# docker volume create portainer_data
portainer_data
```

Maintenant que le volume est créé, nous allons utiliser une commande docker permettant de lancer le container Portainer :

```
docker run -d -p 8000:8000 -p 9443:9443 --name portainer --restart=always -v /var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer-ce:latest
```

```
root@SRVA-DOCKER:~# docker run -d -p 8000:8000 -p 9443:9443 --name portainer --restart=always -v /var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer-ce:latest
Unable to find image 'portainer/portainer-ce:latest' locally
latest: Pulling from portainer/portainer-ce
57654d40e0a5: Pull complete
1f476acfabd6: Pull complete
5171176db7f2: Pull complete
52e9438966a5: Pull complete
43d4775415ac: Pull complete
c1cad9f5200f: Pull complete
a5e2b359b78b: Pull complete
eb172612bcbb: Pull complete
6be7b2acffb5: Pull complete
391dff0fb880: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:4a1ceadd7f7898d9190ee0a6d22234c4323aefd80e796e84f5e57127f74370f1
Status: Downloaded newer image for portainer/portainer-ce:latest
1f745f928050e31cb4fdc60aa80004b0e5cbbfcc1beeecc5317b39525c34180d
```

Si nous analysons les containers en route, nous pouvons voir que notre container portainer est bien démarré

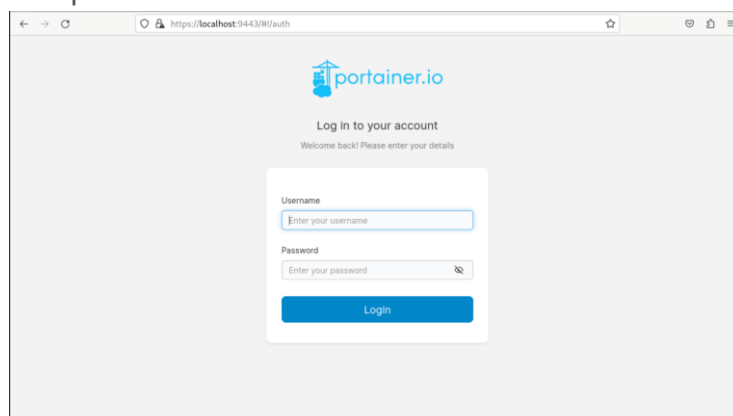
```
root@SRVA-DOCKER:~# docker container ls
```

CONTAINER ID	IMAGE	COMMAND	NAMES	CREATED	STATUS	PORTS
1f745f928050	portainer/portainer-ce:latest	"/portainer"	portainer	6 minutes ago	Up 6 minutes	0.0.0.0:8000->8000/tcp, :::8000->8000/tcp, 0.0.0.0:9443->9443/tcp, :::9443->9443/tcp, 9000/tcp
f71e9a0208ec	nginx:latest	"/docker-entrypoint..."	web-server02	6 weeks ago	Up 6 weeks	0.0.0.0:8082->80/tcp, :::8082->80/tcp
ac4366f5bd8d	nginx:latest	"/docker-entrypoint..."	web-server01	6 weeks ago	Up 6 weeks	0.0.0.0:8081->80/tcp, :::8081->80/tcp

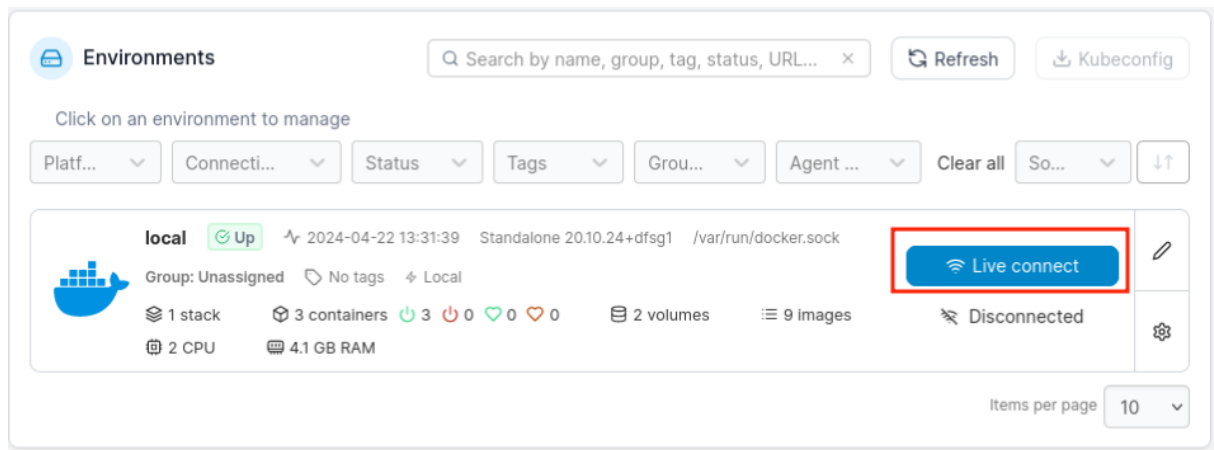
2 – Configuration de portainer

Maintenant que le container est en route nous pouvons accéder à l'interface web en utilisant l'adresse <https://localhost:9443/>

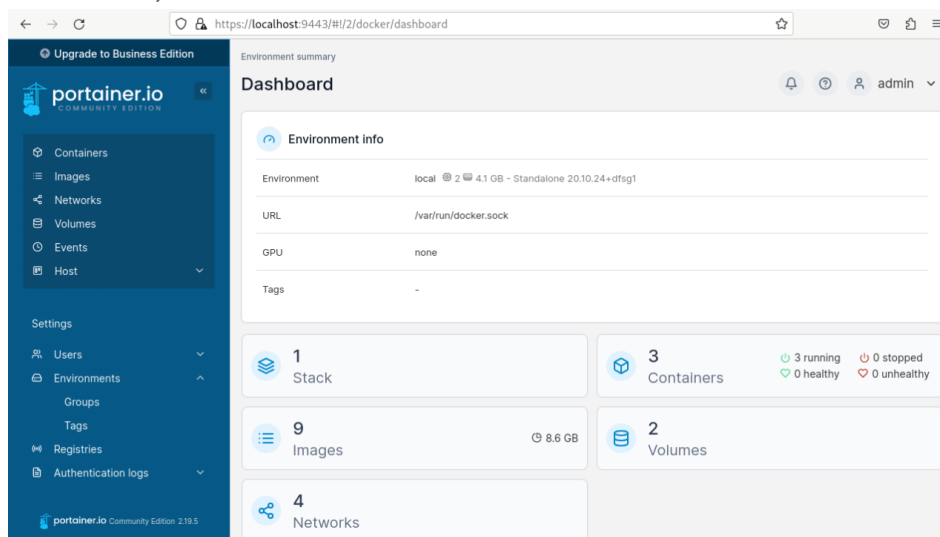
Il vous sera demandé de créer un mot de passe et d'ensuite vous connecter avec le compte admin



Une fois connecté à l'interface il faudra sélectionner notre socket Docker



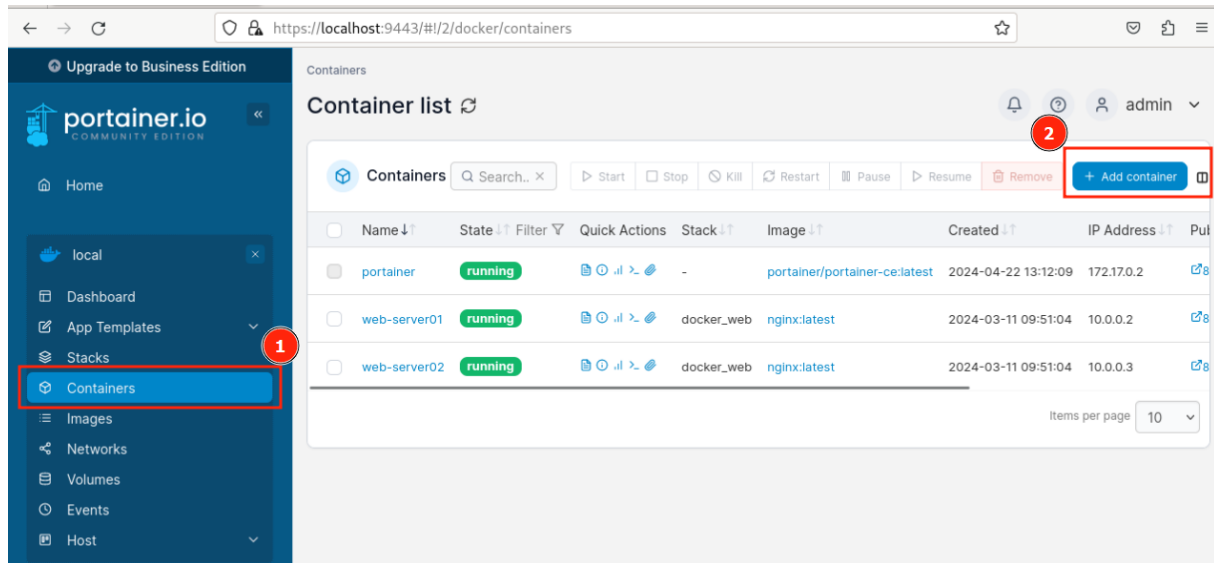
Une fois l'environnement Docker sélectionné, nous avons accès aux images, containers, réseaux et volumes



3 – Créer des containers avec Portainer

Nous allons à présent voir comment créer des containers depuis Portainer, sans utiliser les commandes Docker depuis un terminal.

Rendez-vous dans l'onglet « **Containers** » sur l'interface web et cliquez sur « **Add container** »



Donnez un nom à votre container

Name

container_test

Ensuite, sélectionnez votre image dans le registry de Docker

Image configuration

Registry Docker Hub (anonymous)

Image* docker.io nginx Search

Si cela est nécessaire, il vous est possible de publier des ports

Network ports configuration

Publish all exposed network ports to random host ports ☐

Manual network port publishing ☐ + publish a new network port

Sélectionnez ensuite les utilisateurs pouvant avoir accès au container et cliquez sur « **Deploy the container** »

Access control

Enable access control ☒

☒
Administrators
 I want to restrict the management of this resource to administrators only

☐
Restricted
 I want to restrict the management of this resource to a set of users and/or teams

Actions

Auto remove ☐

Deploy the container

Une fois le container déployé, il devrait apparaître dans la liste de tous les containers

The screenshot shows the Portainer.io web interface. On the left is a sidebar with navigation links: Home, local, Dashboard, App Templates, Stacks, Containers (selected), Images, Networks, Volumes, Events, and Host. The main area is titled 'Container list' and contains a table of containers. The 'container_test' container is highlighted with a red box. Below the table, it says '1 item(s) selected'.

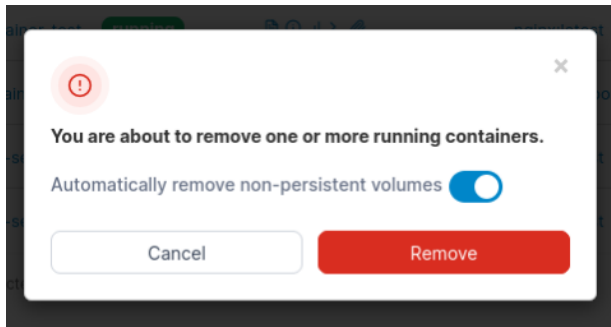
Name	State	Quick Actions	Stack	Image	Created	IP Address
container_test	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	-	nginx:latest	2024-04-22 14:35:41	172.17.0.3
portainer	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	-	portainer/portainer-ce:latest	2024-04-22 13:12:09	172.17.0.2
web-server01	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	docker_web	nginx:latest	2024-03-11 09:51:04	10.0.0.2
web-server02	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	docker_web	nginx:latest	2024-03-11 09:51:04	10.0.0.3

Pour détruire le container, il suffit de le sélectionner et de cliquer sur « **Remove** »

The screenshot shows the Portainer.io web interface with the 'Container list' page. The 'container_test' container is selected, indicated by a red box and a '1' in a red circle. The 'Remove' button is highlighted with a red box and a '2' in a red circle. Below the table, it says '1 item(s) selected'.

Name	State	Quick Actions	Stack	Image	Created	IP Address
container_test	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	-	nginx:latest	2024-04-22 14:35:41	172.17.0.3
portainer	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	-	portainer/portainer-ce:latest	2024-04-22 13:12:09	172.17.0.2
web-server01	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	docker_web	nginx:latest	2024-03-11 09:51:04	10.0.0.2
web-server02	running	[Start] [Stop] [Kill] [Restart] [Pause] [Resume] [Remove]	docker_web	nginx:latest	2024-03-11 09:51:04	10.0.0.3

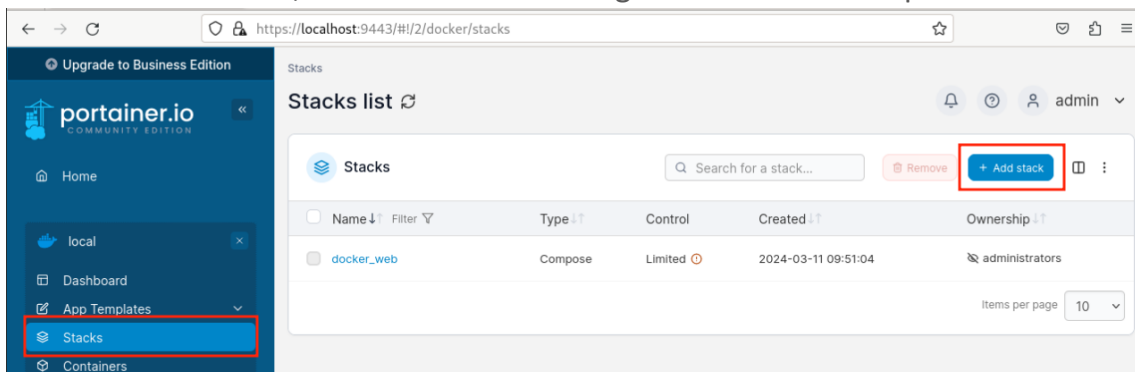
Un message d'avertissement va apparaître, il suffit de cliquer encore une fois sur « Remove »



4 – Onglet Stacks

L'onglet Stacks va permettre de gérer nos fichiers Docker Compose, cependant il est à noter que pour pouvoir modifier une stack depuis portainer, il faut que celle-ci ait été créée depuis portainer.

Pour créer une stack, rendez-vous dans l'onglet « Stacks » et cliquez sur « Add stack »



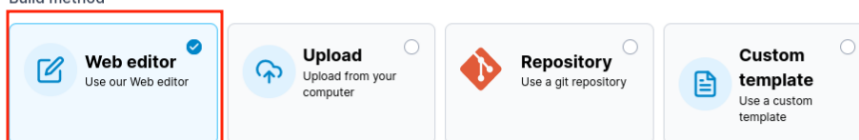
Donnez un nom à celle-ci afin de vous y retrouver par la suite

Name

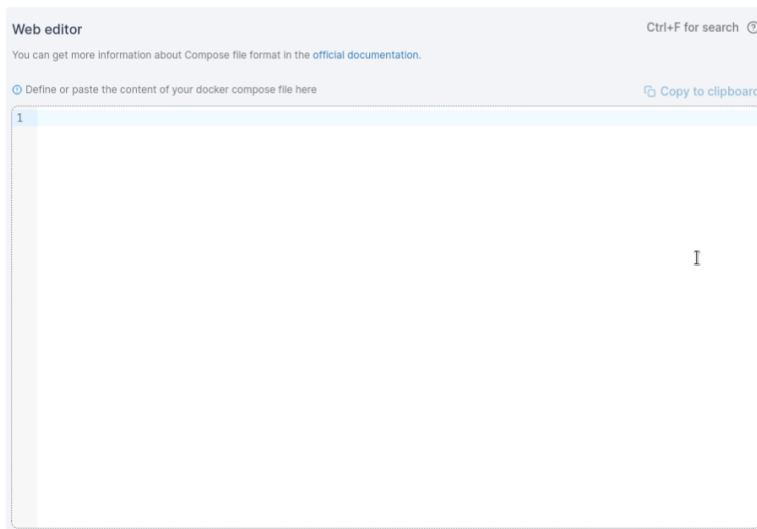
This stack will be deployed using docker compose.

Ensuite, plusieurs options s'offrent à vous pour la création de votre fichier Docker Compose. Personnellement, je préfère écrire les fichiers docker compose moi-même, comme ça je sais réellement ce qu'ils font, il faut donc utiliser la solution « **Web editor** »

Build method



Dans la console située dessous, il ne vous reste plus qu'à écrire votre fichier Docker compose



Comme pour les containers, vous pouvez sélectionner qui aura accès à la stack et les environnements de variable (par exemple si une base de données est utilisée). Une fois terminée il ne vous reste plus qu'à sélectionner « **Deploy the stack** ».

Environment variables

These values will be used as substitutions in the stack file. To reference the .env file in your compose file, use 'stack.env'

[Advanced mode](#)

Switch to advanced mode to copy & paste multiple variables

+ Add an environment variable

Load variables from .env file

Access control

Enable access control ☒

Administrators

I want to restrict the management of this resource to administrators only

Restricted

I want to restrict the management of this resource to a set of users and/or teams

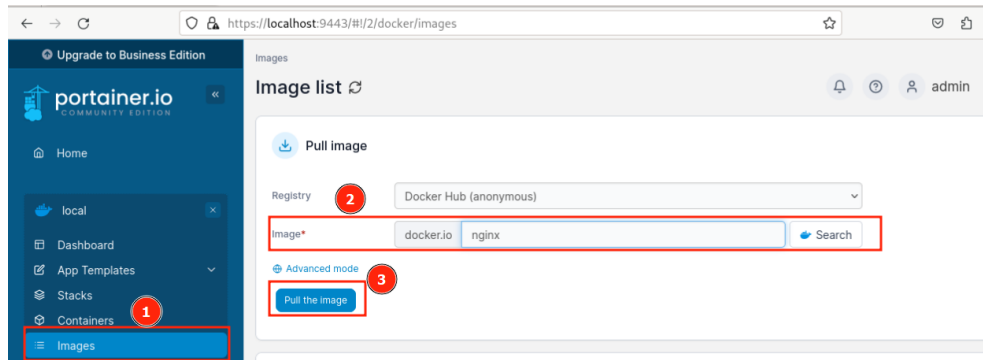
Actions

Deploy the stack

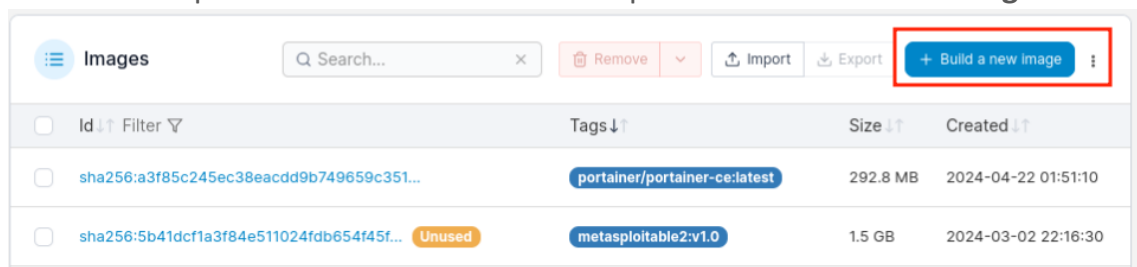
5 – Onglet images

Dans cette partie de portainer, nous pouvons pull des images disponible dans le registry Docker, ou bien créer notre propre image Docker File

Pour Pull une image, rendez-vous dans « **Images** », sélectionnez l'image et cliquez sur « **Pull** »



Il est également possible, comme dis précédemment, de créer sa propre image avec Dockerfile depuis Portainer. Pour ce faire cliquez sur « **Build a new image** »



Donnez ensuite un nom à votre image en suivant la convention donnée par portainer
Naming

You can specify multiple names to your image.

Names + add additional name

A name must be specified in one of the following formats: `name:tag`, `repository/name:tag` or `registryfqdn:port/repository/name:tag` format. If you omit the tag the default **latest** value is assumed.

name × 🗑️

⚠ The image name must consist of between 2 and 255 lowercase alphanumeric characters, '-', '_' or '.' (e.g. 'my-name', or 'abc-123').

Ensuite écrivez votre Dockerfile à l'aide de la console

Web editor

You can get more information about Dockerfile format in the [official documentation](#).

Define or paste the content of your Dockerfile here

[Copy to clipboard](#)

1

Une fois votre fichier écrit, il ne vous reste plus qu'à cliquer sur « **Build the image** »

Upload

You can upload files from your local computer for referencing in your Dockerfile (using ADD filename) so they are included in your built image.

Select files

Actions

Build the image

6 – Onglet Networks

Dans cet onglet, nous allons pouvoir gérer tous nos réseau Docker

Name	Stack	Driver	Attachable	IPAM Driver	IPv4 IPAM Subnet	IPv4 IPAM Gateway
bridge	-	bridge	false	default	172.17.0.0/16	172.17.0.1
docker_web_web_servers	docker_web	bridge	true	default	10.0.0.0/24	-
host	-	host	false	default	-	-
none	-	null	false	default	-	-

Pour pouvoir créer un nouveau réseau Docker, cliquez sur « **Add network** »

Il va falloir en premier lieu donner un nom à votre réseau et sélectionner le type de réseau

Name

Driver configuration

Driver

Driver options ⓘ

+ Add driver option

Vous allez maintenant pouvoir définir la plage de votre réseau ainsi qu'une passerelle

IPv4 Network configuration

Subnet Gateway

IP range

+ Add excluded IP

Ensuite, vous pouvez choisir dans les options avancées d'isoler le réseau ou non et qui a accès à ce réseau. Une fois tout cela fait, cliquez sur « **Create the network** »

Advanced configuration

+ Add label

Isolated network ☒

Enable manual container attachment ☐

Access control

Enable access control ☒

☒
Administrators
 I want to restrict the management of this resource to administrators only

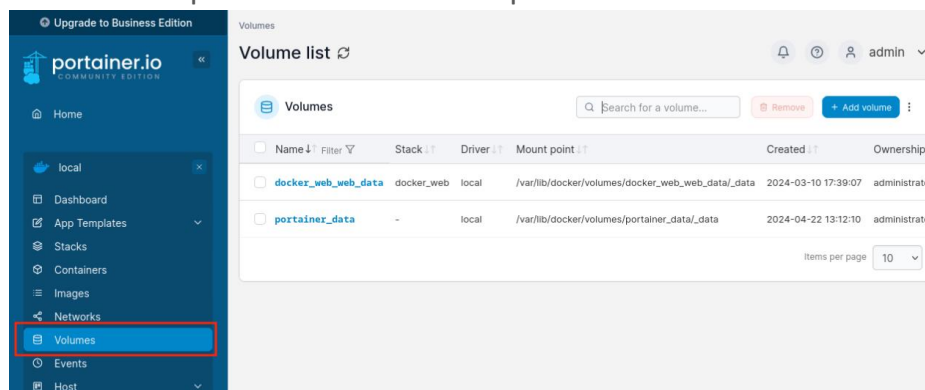
☐
Restricted
 I want to restrict the management of this resource to a set of users and/or teams

Actions

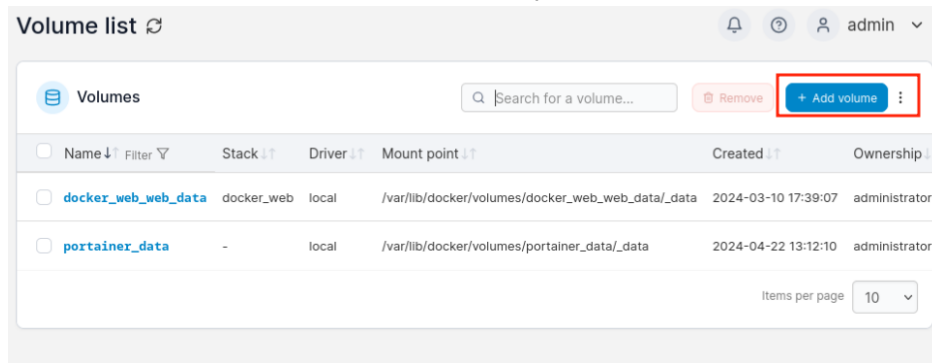
Create the network

7 – Onglet Volumes

Dans cet onglet, vous allez pouvoir créer des volumes Docker. Pour y accéder rendez-vous dans la partie « **Volumes** » de portainer



Afin de créer un nouveau volume, cliquez sur « **Add volume** »



Donnez un nom à votre volume et sélectionnez si vous souhaitez monter ce volume vers un chemin ou juste créer un volume docker qui pourra être utilisé plus tard par des containers

Name

Driver configuration

Driver

Driver options + add driver option

name	value
e.g. mountpoint	e.g. /path/on/host

Use NFS volume ☐

Use CIFS volume ☐

Access control

Enable access control ☒

Administrators
I want to restrict the management of this resource to administrators only

☒

Restricted
I want to restrict the management of this resource to a set of users and/or teams

☐

Actions

Fin de la procédure.