

Installation MQTT sur Docker

Installation du broker Mosquitto sur Docker

<https://technologie-geek.fr/installation-broker-mosquitto-docker/> 1 janvier 2024 par Stéphane

Dernière mise à jour le 24 mars 2024 Installation du broker Mosquitto sur docker Découvrez notre tutoriel d'installation du broker MQTT Mosquitto sur Docker !



Mosquitto c'est quoi ?

<https://mosquitto.org/>

Pour commencer, bonjour et bienvenue sur mon blog Technologie Geek ! Voyons ensemble l'installation du broker MQTT Mosquitto sur Docker. En premier lieu Mosquitto est un broker MQTT open source pour la communication machine à machine (M2M) et l'Internet des objets (IoT). Ensuite il permet aux appareils de communiquer entre eux de manière efficace et fiable.

De même avec Mosquitto, vous pouvez mettre en place des réseaux d'appareils IoT et les faire communiquer entre eux de manière sécurisée. Enfin Mosquitto prend en charge les protocoles MQTT et MQTT-SN pour la communication à faible consommation d'énergie.

En outre, Mosquitto est facile à installer et à configurer. Par ailleurs il peut être utilisé sur une variété de plateformes. Ordinateurs de bureau, les serveurs et les périphériques embarqués. Il est compatible avec les langages de programmation populaires tels que C, C++, Python et Java.

En sommes, Mosquitto offre des fonctionnalités avancées pour la gestion des messages MQTT. Rétention des messages, la publication et la souscription à des thèmes, la sécurité de la communication et la mise en place de ponts entre les brokers.

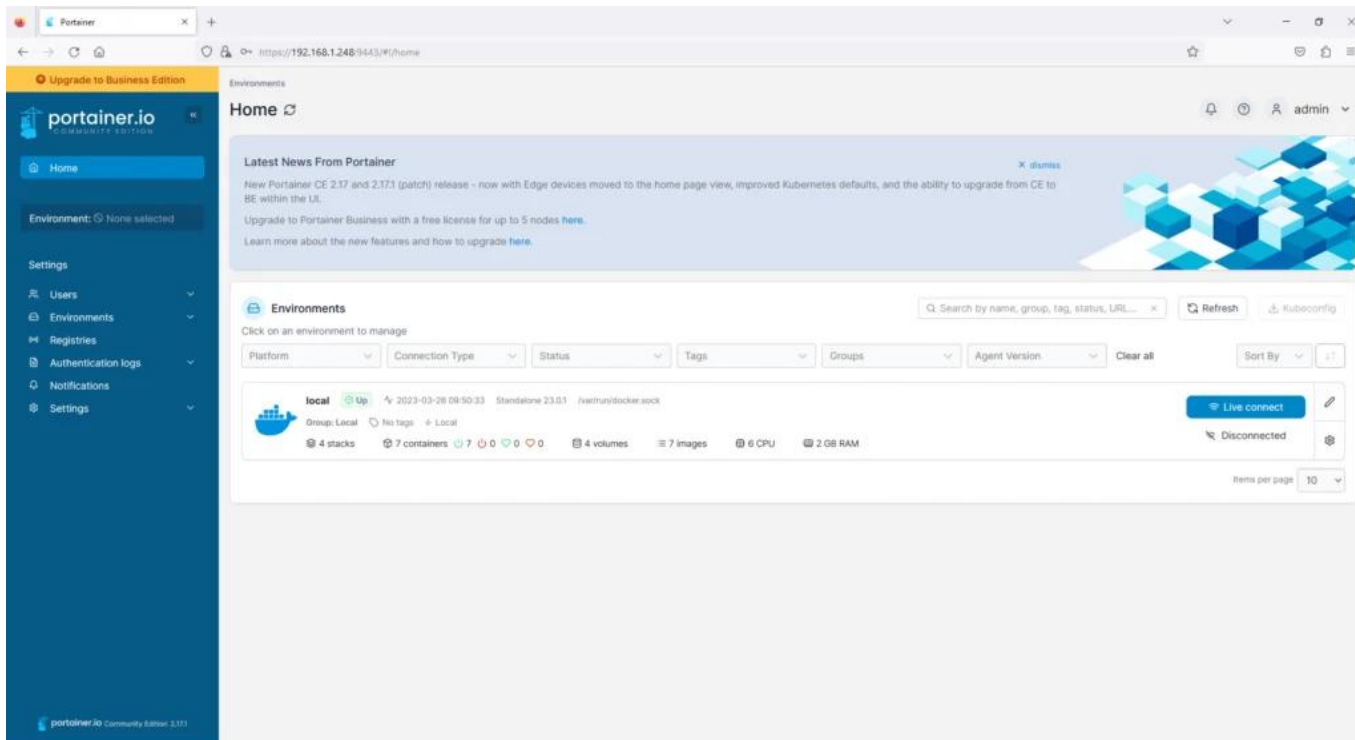
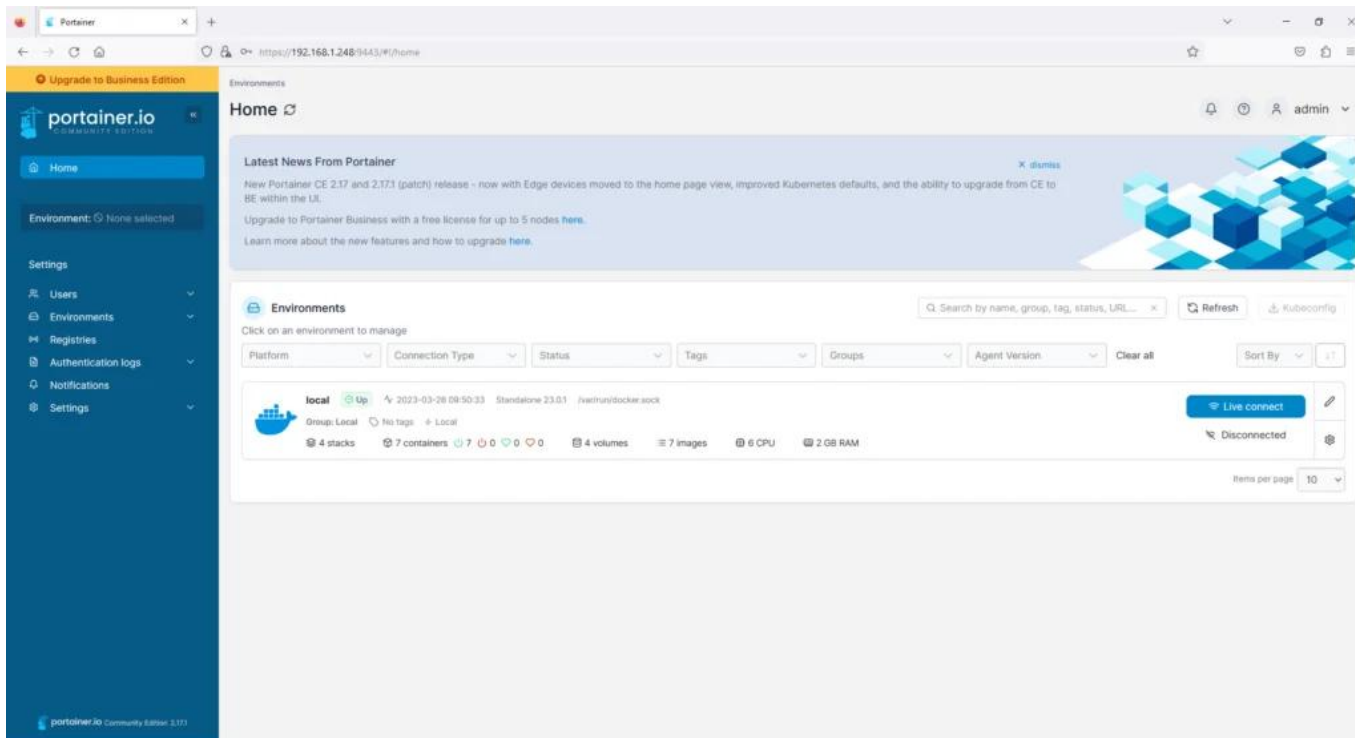
En outre, Mosquitto est largement utilisé dans les applications IoT, y compris la maison intelligente, la gestion de l'énergie, la santé connectée, les villes intelligentes et l'agriculture intelligente.

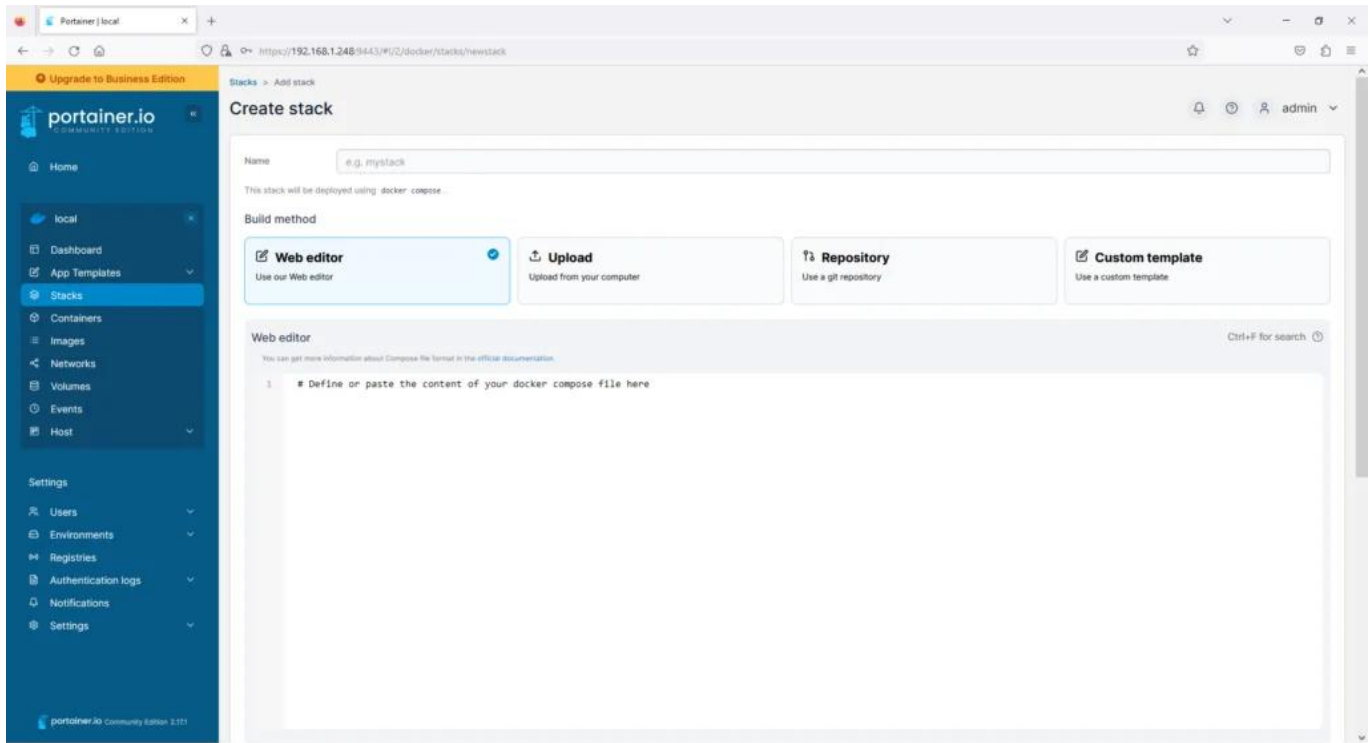
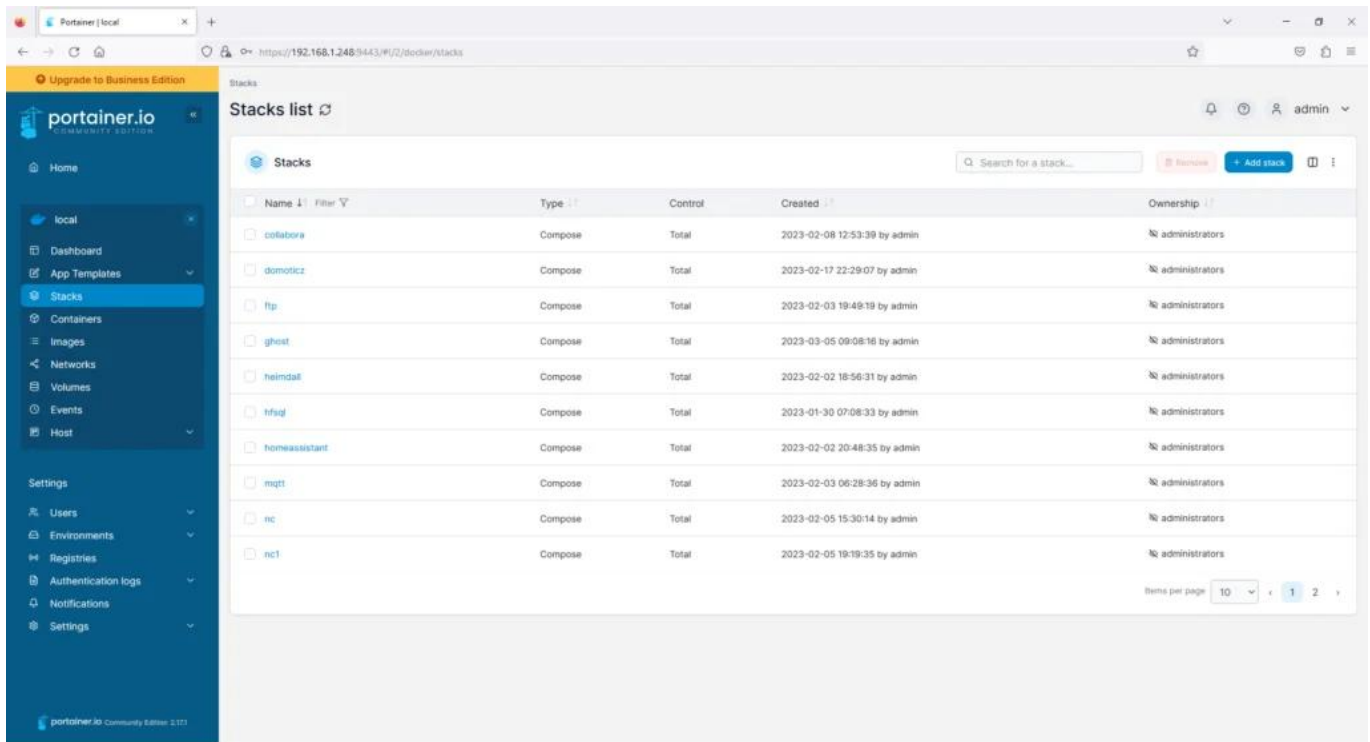
En résumé, Mosquitto est un broker MQTT efficace et facile à utiliser pour la communication machine à machine et l'Internet des objets. Si vous êtes un développeur ou une entreprise cherchant à implémenter des réseaux IoT, Mosquitto est certainement une solution à considérer. Prérequis pour l'installation de Mosquitto sur Docker :

1. En premier lieu une installation de Debian.
2. Ensuite une installation de Docker.
3. Enfin Portainer installer sur Docker pour l'utilisation du format docker-compose.

Installation du broker MQTT Mosquitto sur Docker :

1. Pour débuter, aller sur la page d'administration de Portainer.
2. Ensuite, sélectionner votre environnement « local ».
3. Puis, cliquer sur « Stacks ».
4. Finalement, cliquer sur « add Stack ».





Installation du broker MQTT Mosquitto sur Docker

Ajouter stack

1. Premièrement, dans « Name » taper le nom de votre Stack. Ici ce sera « mqtt ».
1. Puis, dans la partie « Web editor », coller le code ci-dessous :

[mqttdocker001.txt](#)

```
version: "3.8"
services:
  mosquitto:
    image: eclipse-mosquitto
    container_name: mqtt
    restart: always
    volumes:
      - config:/mosquitto/config
      - data:/mosquitto/data
      - log:/mosquitto/log
    ports:
      - 1883:1883
      - 9001:9001
volumes:
  config:
  data:
  log:
```

1. Ensuite, pour déployer votre broker Mosquitto sur Docker, en bas de la page appuyer sur le bouton « Deploy the stack », puis patienter.

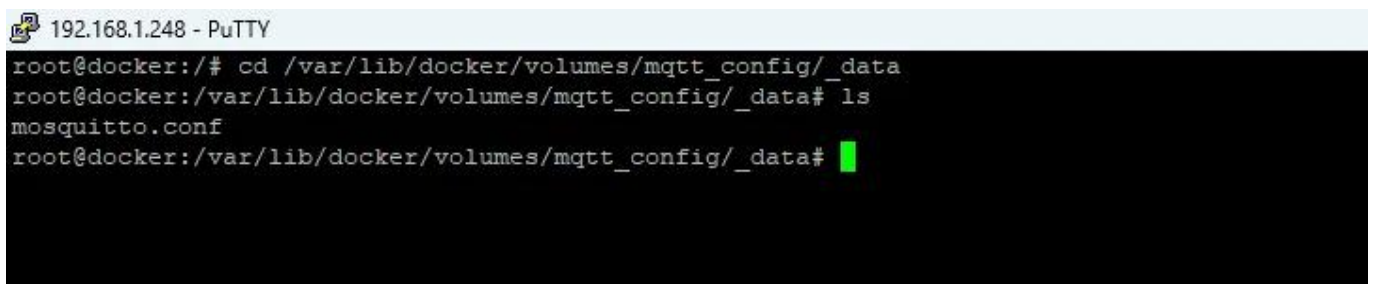
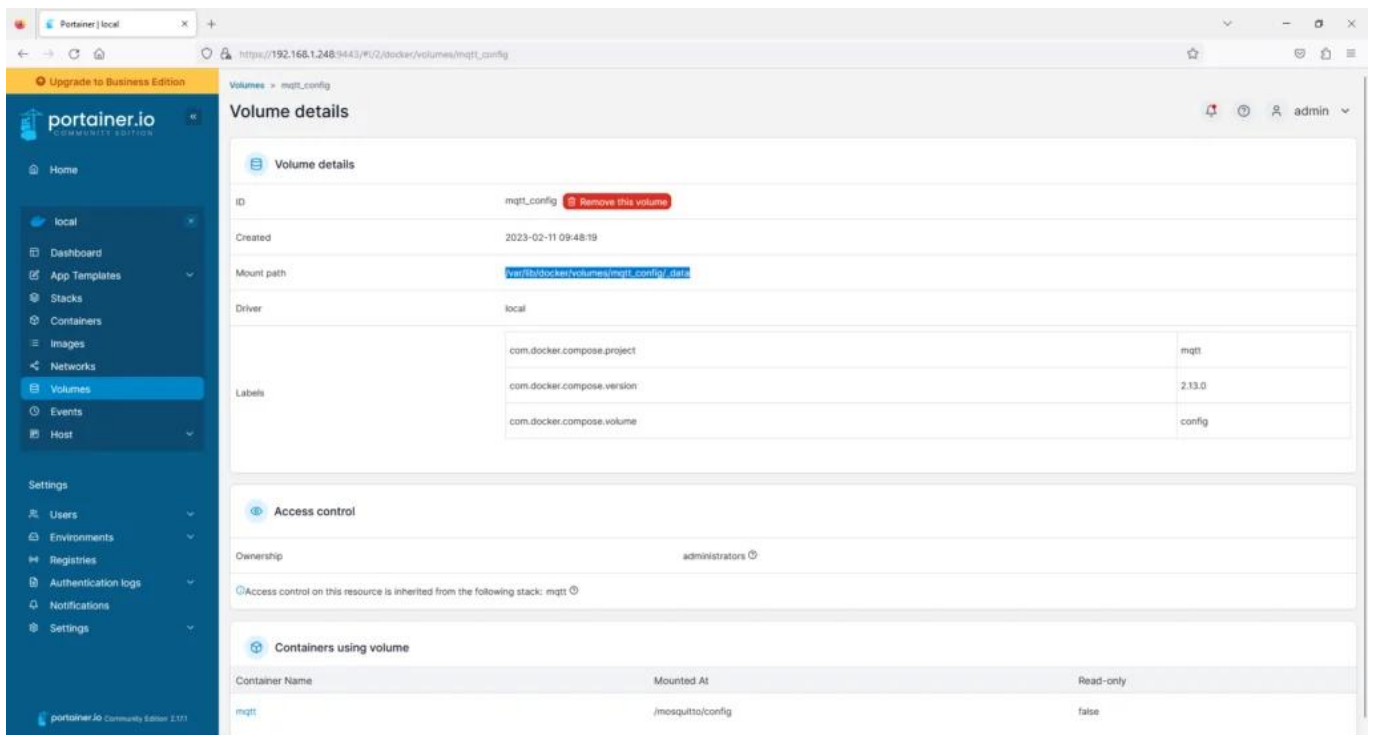
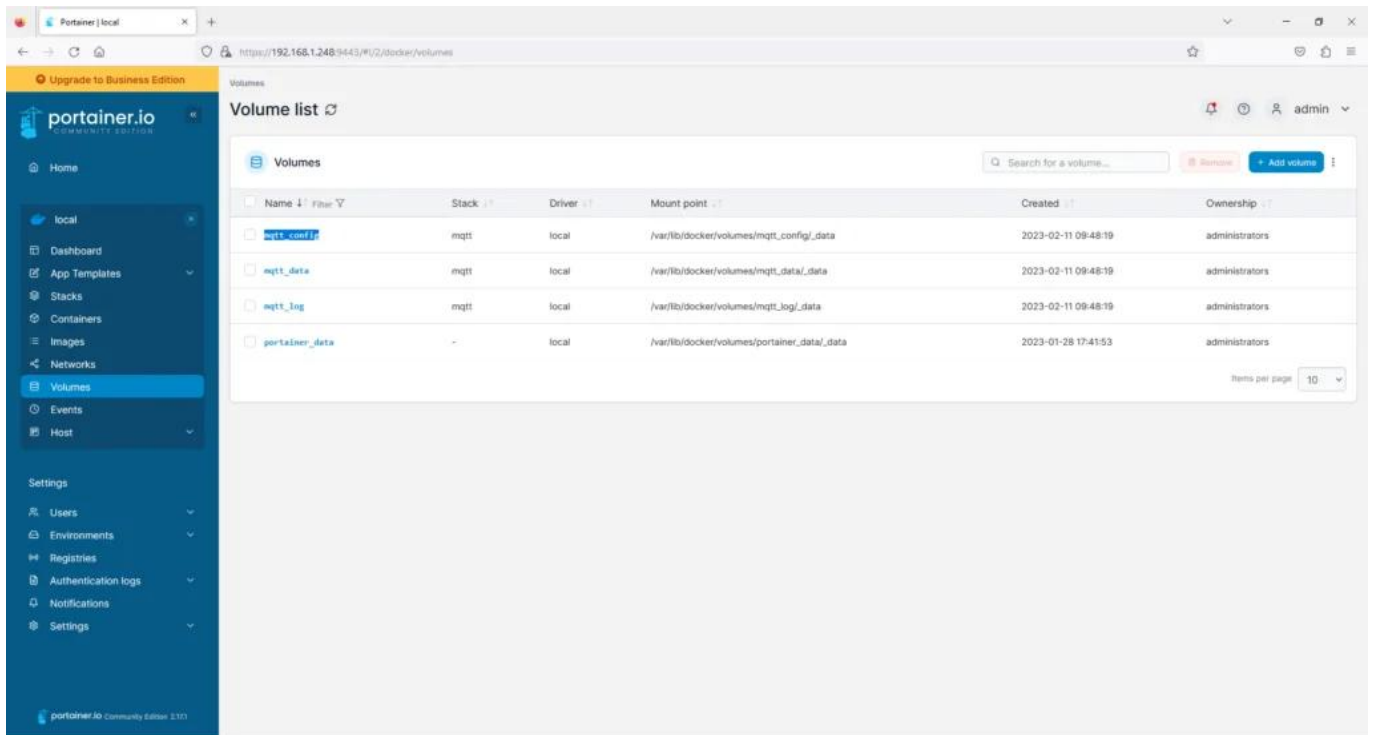
Explication du code :

- Version: '3.8' est la version du moteur Docker. Connectez vous en ssh et taper « docker -v ». En suite pour plus d'info aller à page Docker versioning.
1. services: Démarre un nouveau service.
 2. mosquitto: Est le nom du service.
 3. image: Est le lien vers l'image utilisée. Rendez-vous sur Docker hub.
 4. container_name: Est le nom du conteneur.
 5. restart: Lorsque le conteneur est stoppé, nous le redémarrons toujours.
 6. volumes: Sont vos volumes pour la persistance des données.
 7. ports: Nous exposons les ports 1883 et 9001 de notre conteneur.

Création d'un utilisateur :

1. Maintenant que votre conteneur fonctionne, vous devez créer un utilisateur et un mot de passe. Pour cela connectez vous en ssh sur notre serveur.
1. Rendez vous dans le volume « config » de votre conteneur. Vous trouverez le chemin de votre volume sur Portainer dans l'onglet « volume » puis cliquez sur le nom du volume.

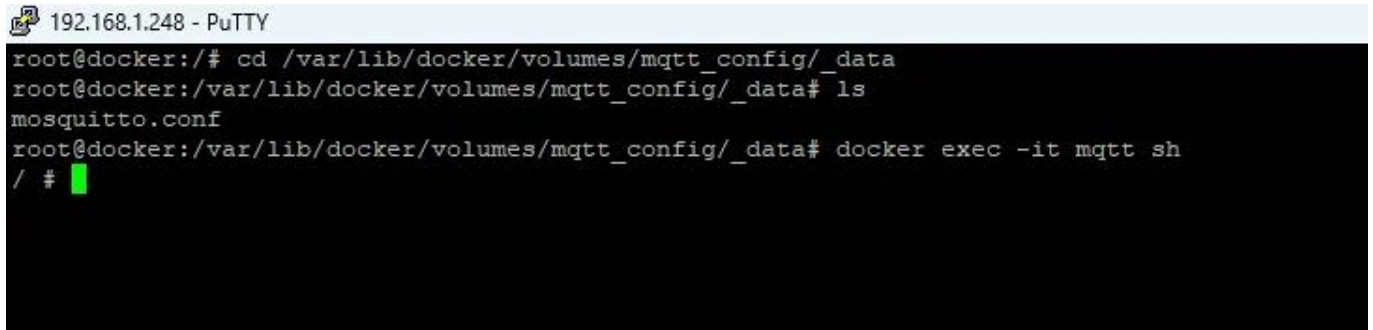
```
cd /var/lib/docker/volumes/mqtt_config/_data
```



installation du broker MQTT Mosquitto sur Docker

1. Ensuite, connectez vous en ssh sur votre Installation du broker Mosquitto sur Docker.

```
docker exec -it mqtt sh
```



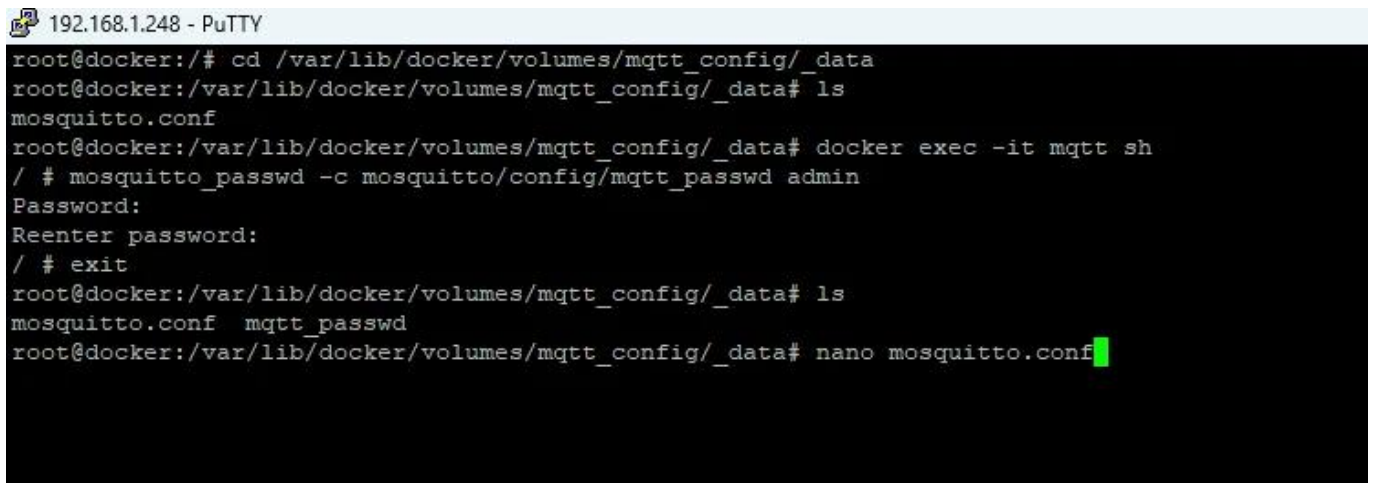
```
192.168.1.248 - PuTTY
root@docker:/# cd /var/lib/docker/volumes/mqtt_config/_data
root@docker:/var/lib/docker/volumes/mqtt_config/_data# ls
mosquitto.conf
root@docker:/var/lib/docker/volumes/mqtt_config/_data# docker exec -it mqtt sh
/ #
```

-Puis, vous devez créer votre utilisateur « admin » et définir son mot de passe.

```
mosquitto_passwd -c mosquitto/config/mqtt_passwd admin
```

-Vous pouvez taper exit pour sortir de la console ssh de votre conteneur. Ensuite, éditer le fichier « mosquitto.conf » dans le volume « config » avec le code suivant.

```
nano mosquitto.conf
```



```
192.168.1.248 - PuTTY
root@docker:/# cd /var/lib/docker/volumes/mqtt_config/_data
root@docker:/var/lib/docker/volumes/mqtt_config/_data# ls
mosquitto.conf
root@docker:/var/lib/docker/volumes/mqtt_config/_data# docker exec -it mqtt sh
/ # mosquitto_passwd -c mosquitto/config/mqtt_passwd admin
Password:
Reenter password:
/ # exit
root@docker:/var/lib/docker/volumes/mqtt_config/_data# ls
mosquitto.conf mqtt_passwd
root@docker:/var/lib/docker/volumes/mqtt_config/_data# nano mosquitto.conf
```

Edition config

1. puis, pour terminer l'installation du broker Mosquitto sur Docker collez le code ci-dessous.

[mqttdocker002.txt](#)

```
listener 1883
listener 9001
protocol websockets
```

```
persistence true
persistence_location /mosquitto/data
allow_anonymous false
password_file mosquitto/config/mqtt_passwd
```

Installation du broker MQTT Mosquitto sur Docker

1. En dernier lieu, sauvegardé votre fichier et redémarré votre conteneur. Pour tester votre broker Mosquitto, vous pouvez télécharger MQTT Explorer et essayer de vous connecter.

From:

<https://www.magenealogie.chanterie37.fr/www/fablab37110/> - Castel'Lab le Fablab MJC de Château-Renault

Permanent link:

<https://www.magenealogie.chanterie37.fr/www/fablab37110/doku.php?id=start:raspberry:mqtt:docker&rev=1770110822>

Last update: **2026/02/03 10:27**

