

Installation, configuration et utilisation de Docker.

Clément LAMPS, Ecoris Annecy



Table des matières

SUIVI DES VERSIONS	5
PREREQUIS	6
MISE A JOUR DU SYSTEME.....	6
INSTALLATION DE DOCKER.....	7
AJOUT DU DEPOT	7
INSTALLATION DE DOCKER.....	7
TEST DE DOCKER.....	9
INSTALLATION DE PORTAINER	10
INSTALLATION DE PORTAINER	10
UTILISATION DE PORTAINER.....	13
INSTALLATION DE NGINX PROXY MANAGER	14
CREATION DU FICHIER COMPOSE	14

Installation, configuration et utilisation de Docker

LANCEMENT DU CONTAINER

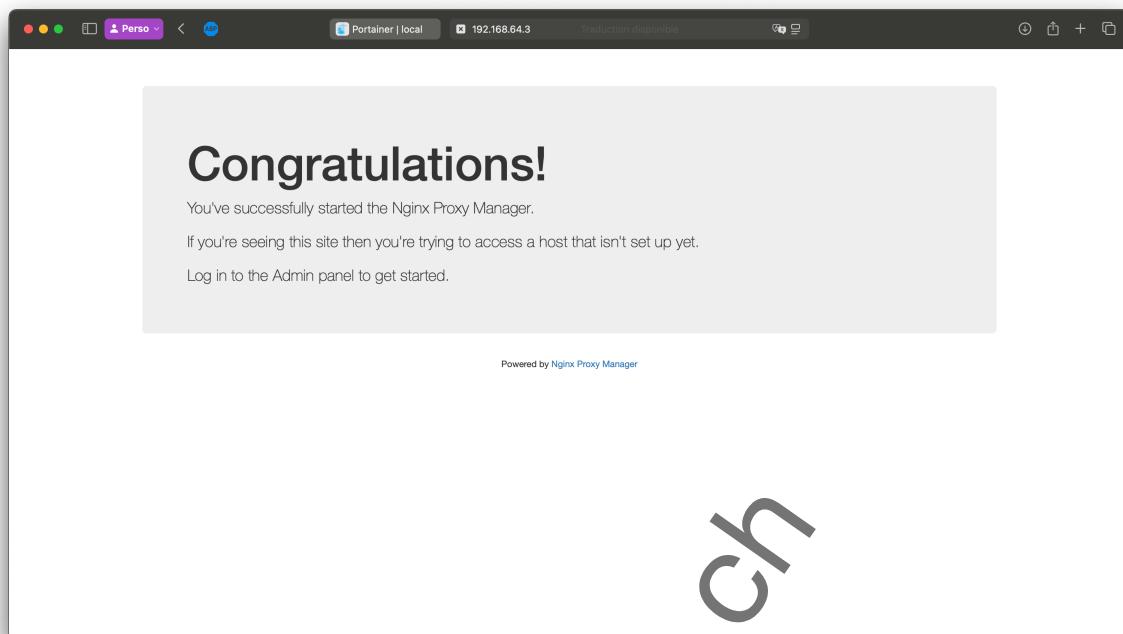
```
[[pressynou@localhost ~]$ sudo docker compose up -d
[+] Running 3/36
  :: app [ . . . ] Pulling 5.6s
```

15

Name	State	Stack	Image	Created	IP Address	Published Ports	Own
portainer	running	-	portainer/portainer-ce:2.21.5	2025-02-11 08:30:53	172.17.0.2	8443:9443 8000:8000	a
nginx-proxy-app-1	running	pressynou/jc21/nginx-proxy-manager:latest		2025-02-11 08:41:41	172.18.0.2	8443:443 80:80 81:81	a

CONNEXION A NGINX PROXY MANAGER

Installation, configuration et utilisation de Docker



16

Suivi des versions

Version	Date	Rédacteur	Modification
V1.0	10/02/2025	Clément Lamps	Creation du document
V1.0	14/02/2025	CLEMENT LAMPS	AJOUT DE PORTAIENR

clementlamps.ch

Prérequis



Ce tuto est rédigé pour un système ARM, la méthodologie d'installation peut légèrement différer.

Il est nécessaire d'avoir un serveur linux fonctionnel et à jour. Ce tuto s'adresse aux distributions dérivées du système RHEL. Il faut aussi avoir les permissions d'administrateur (sudo), ou être connecter au compte root

Pour cette procédure, j'ai utilisé une machine virtuelle en ARM d'un Rocky Linux 9.

Mise à jour du système

Pour mettre à jour le système afin d'avoir les derniers correctifs et paquets, on exécute DNF

1. sudo dnf update -y
2. sudo dnf upgrade -yup

Vous devrez avoir un résultat comme celui-ci

The screenshot shows a terminal window with the following output:

```
pressynou — root@localhost:~— ssh root@192.168.64.3 — 80x24

Vérification de : sos-4.8.2-2.el9_5.noarch 9/18
Vérification de : sos-4.8.1-1.el9_5.noarch 10/18
Vérification de : libstdc++-11.5.0-5.el9_5.aarch64 11/18
Vérification de : libstdc++-11.5.0-2.el9.aarch64 12/18
Vérification de : libgomp-11.5.0-5.el9_5.aarch64 13/18
Vérification de : libgcc-11.5.0-2.el9.aarch64 14/18
Vérification de : libgcc-11.5.0-5.el9_5.aarch64 15/18
Vérification de : libatomic-11.5.0-2.el9.aarch64 16/18
Vérification de : libatomic-11.5.0-5.el9_5.aarch64 17/18
Vérification de : libatomic-11.5.0-2.el9.aarch64 18/18

Mis à niveau:
libatomic-11.5.0-5.el9_5.aarch64
libgcc-11.5.0-5.el9_5.aarch64
libgomp-11.5.0-5.el9_5.aarch64
libstdc++-11.5.0-5.el9_5.aarch64
libxml2-2.9.13-6.el9_5.1.aarch64
openssl-1:3.1.2-1.el9_5.1.aarch64
openssl-libs-1:3.1.2-6.el9_5.1.aarch64
python3-libxml2-2.9.13-6.el9_5.1.aarch64
sos-4.8.2-2.el9_5.noarch

Terminé !
[root@localhost ~]#
```

Installation de docker

Docker n'est pas disponible dans les dépôts par default de Rocky Linux, nous allons donc ajouter son dépôt.

Ajout du dépôt

Ajoutez le dépôt Docker officiel :

```
1. sudo dnf config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo
```

The screenshot shows a terminal window titled "pressynou — root@localhost:~ — ssh root@192.168.64.3 — 80x24". The terminal displays the following text:

```
Last login: Sun Feb  9 15:41:07 on ttys000
[pressynou@MacBookPro-De-Clement ~ % ssh root@192.168.64.3
The authenticity of host '192.168.64.3 (192.168.64.3)' can't be established.
ED25519 key fingerprint is SHA256:W+lbYnI5Z0FaXf5EgMWxiTwhf4ZIR0bqbYtbp5qtkw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no, [fingerprint])? yes
Warning: Permanently added '192.168.64.3' (ED25519) to the list of known hosts.
[root@192.168.64.3's password:
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Tue Feb 11 08:26:16 2025
[[root@localhost ~]# dnf config-manager --add-repo https://download.docker.com/linux/rhel/docker-ce.repo
Ajout du dépôt depuis : https://download.docker.com/linux/rhel/docker-ce.repo
[root@localhost ~]# ]
```

Installation de Docker

Installez Docker et ses dépendances :

```
1. sudo dnf install -y docker-ce docker-ce-cli containerd.io
```

Installation, configuration et utilisation de Docker

```
pressynou — root@localhost:~ — ssh root@192.168.64.3 — 80x24

Exécution du scriptlet: docker-ce-cli-1:27.5.1-1.el9.aarch64      3/6
Installation       : containerd.io-1.7.25-3.1.el9.aarch64          4/6
Exécution du scriptlet: containerd.io-1.7.25-3.1.el9.aarch64      4/6
Installation       : docker-ce-rootless-extras-27.5.1-1.el9.aarch64  5/6
Exécution du scriptlet: docker-ce-rootless-extras-27.5.1-1.el9.aarch64  5/6
Installation       : docker-ce-3:27.5.1-1.el9.aarch64              6/6
Exécution du scriptlet: docker-ce-3:27.5.1-1.el9.aarch64              6/6
Vérification de   : containerd.io-1.7.25-3.1.el9.aarch64          1/6
Vérification de   : docker-buildx-plugin-0.20.0-1.el9.aarch64        2/6
Vérification de   : docker-ce-3:27.5.1-1.el9.aarch64              3/6
Vérification de   : docker-ce-cli-1:27.5.1-1.el9.aarch64          4/6
Vérification de   : docker-ce-rootless-extras-27.5.1-1.el9.aarch64  5/6
Vérification de   : docker-compose-plugin-2.32.4-1.el9.aarch64        6/6

Installé:
containerd.io-1.7.25-3.1.el9.aarch64
docker-buildx-plugin-0.20.0-1.el9.aarch64
docker-ce-3:27.5.1-1.el9.aarch64
docker-ce-1:27.5.1-1.el9.aarch64
docker-ce-rootless-extras-27.5.1-1.el9.aarch64
docker-compose-plugin-2.32.4-1.el9.aarch64

Terminé !
[root@localhost ~]#
```

Activation et démarrage du service Docker :

```
1. sudo systemctl enable --now docker
```

```
pressynou — root@localhost:~ — ssh root@192.168.64.3 — 80x24

Installation       : docker-ce-rootless-extras-27.5.1-1.el9.aarch64  5/6
Exécution du scriptlet: docker-ce-rootless-extras-27.5.1-1.el9.aarch64  5/6
Installation       : docker-ce-3:27.5.1-1.el9.aarch64              6/6
Exécution du scriptlet: docker-ce-3:27.5.1-1.el9.aarch64              6/6
Vérification de   : containerd.io-1.7.25-3.1.el9.aarch64          1/6
Vérification de   : docker-buildx-plugin-0.20.0-1.el9.aarch64        2/6
Vérification de   : docker-ce-3:27.5.1-1.el9.aarch64              3/6
Vérification de   : docker-ce-cli-1:27.5.1-1.el9.aarch64          4/6
Vérification de   : docker-ce-rootless-extras-27.5.1-1.el9.aarch64  5/6
Vérification de   : docker-compose-plugin-2.32.4-1.el9.aarch64        6/6

Installé:
containerd.io-1.7.25-3.1.el9.aarch64
docker-buildx-plugin-0.20.0-1.el9.aarch64
docker-ce-3:27.5.1-1.el9.aarch64
docker-ce-1:27.5.1-1.el9.aarch64
docker-ce-rootless-extras-27.5.1-1.el9.aarch64
docker-compose-plugin-2.32.4-1.el9.aarch64

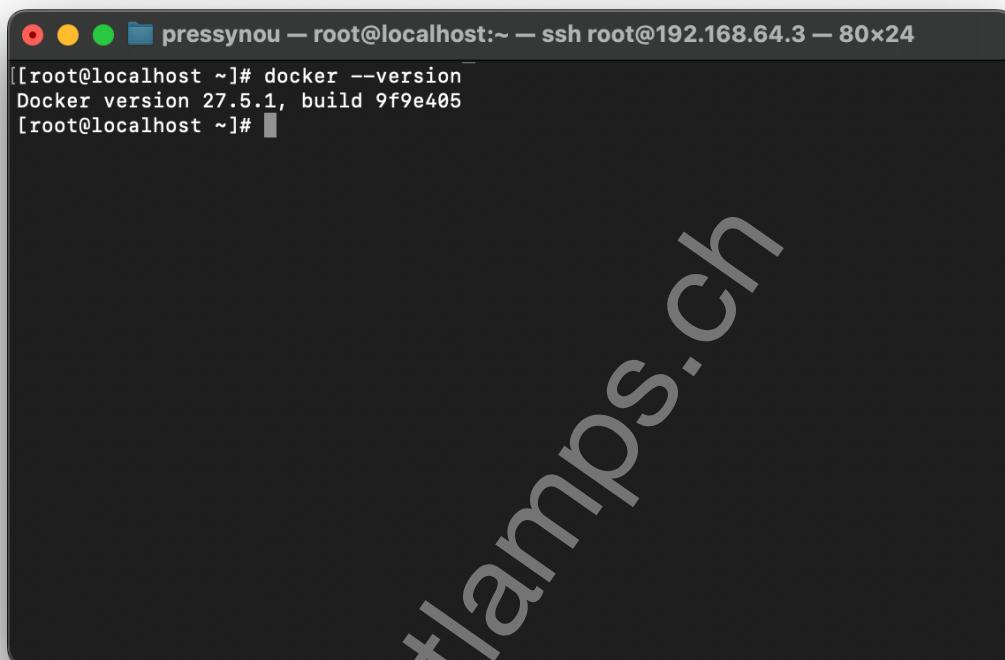
Terminé !
[[root@localhost ~]# systemctl --now enable docker
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
[root@localhost ~]#]
```

Test de Docker

Vérification de installation:

```
1. docker --version
```

Vous devrez avoir un résultat semblable à celui-ci :



The screenshot shows a terminal window with a dark background and light-colored text. At the top, there are three small colored icons (red, yellow, green) followed by the text "pressynou — root@localhost:~ — ssh root@192.168.64.3 — 80x24". Below this, the command "docker --version" is entered, followed by its output: "Docker version 27.5.1, build 9f9e405". The prompt "[root@localhost ~]#" is shown at the bottom.

```
pressynou — root@localhost:~ — ssh root@192.168.64.3 — 80x24
[[root@localhost ~]# docker --version
Docker version 27.5.1, build 9f9e405
[root@localhost ~]# ]
```

Installation de portainer

Installation de portainer

```
pressynou — root@localhost:~ — ssh root@192.168.64.3 — 80x24
Installation : docker-ce-3:27.5.1-1.el9.aarch64 6/6
Exécution du scriptlet: docker-ce-3:27.5.1-1.el9.aarch64 6/6
Vérification de : containerd.io-1.7.25-3.1.el9.aarch64 1/6
Vérification de : docker-buildx-plugin-0.20.0-1.el9.aarch64 2/6
Vérification de : docker-ce-3:27.5.1-1.el9.aarch64 3/6
Vérification de : docker-ce-cli-1:27.5.1-1.el9.aarch64 4/6
Vérification de : docker-ce-rootless-extras-27.5.1-1.el9.aarch64 5/6
Vérification de : docker-compose-plugin-2.32.4-1.el9.aarch64 6/6

Installé:
containerd.io-1.7.25-3.1.el9.aarch64
docker-buildx-plugin-0.20.0-1.el9.aarch64
docker-ce-3:27.5.1-1.el9.aarch64
docker-ce-cli-1:27.5.1-1.el9.aarch64
docker-ce-rootless-extras-27.5.1-1.el9.aarch64
docker-compose-plugin-2.32.4-1.el9.aarch64

Terminé !
[[root@localhost ~]# systemctl --now enable docker
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
[[root@localhost ~]# docker volume create portainer_data
portainer_data
[root@localhost ~]# ]
```

```
1. docker volume create portainer_data
```

```
1. docker run -d \
--name portainer \
--restart=unless-stopped
-p 8000:8000 \
-p 9000:9000 \
-p 9443:9443 \
-v /var/run/docker.sock:/var/run/docker.sock \
-v portainer_data:/data \
portainer/portainer-ce:latest10.
```

Installation, configuration et utilisation de Docker

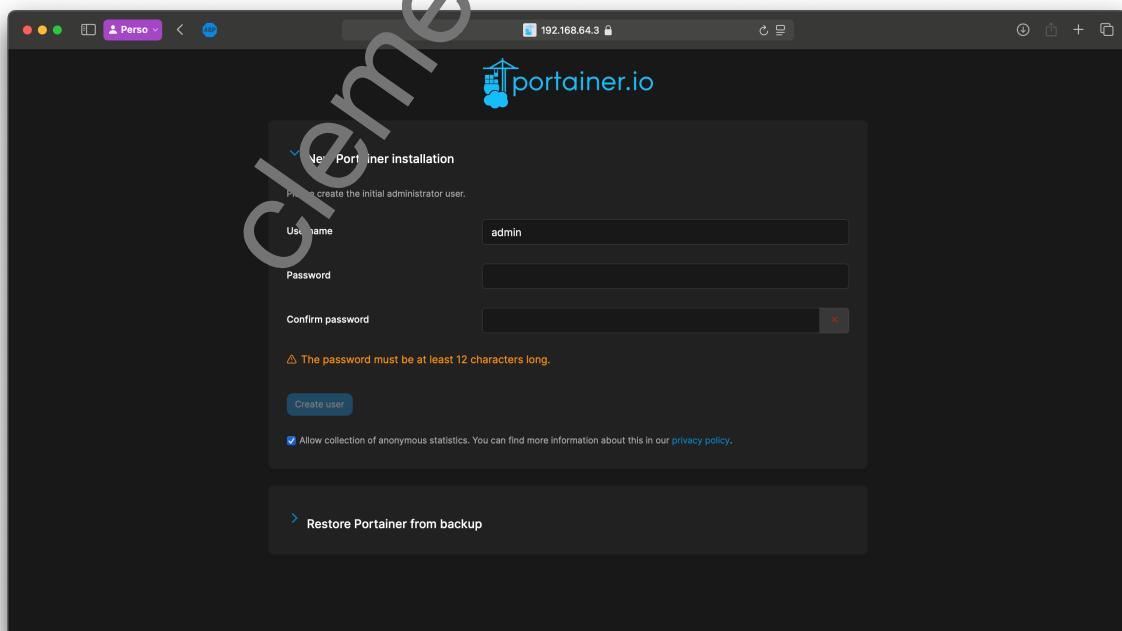
On démarre le container :

```
1. docker-compose up -d
```

```
pressynou — root@localhost:~ — ssh root@192.168.64.3 — 80x24
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
[[root@localhost ~]# docker volume create portainer_data
portainer_data
[root@localhost ~]# 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:2.21.5
Unable to find image 'portainer/portainer-ce:2.21.5' locally
2.21.5: Pulling from portainer/portainer-ce
dc8df0f2921e: Pull complete
c82aa9c9fb45: Pull complete
39b5457baa4a: Pull complete
98965ff91f3e: Pull complete
c1d9734a94ea: Pull complete
2eba1d1ead55: Pull complete
0d49f1fb546f: Pull complete
ce3fb5052b7f: Pull complete
055532548445: Pull complete
793e77bf062e: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:bd8f7a6d98e2a512e18272c38914ab11e9.d663451f3c925d502a8557a3b92d7
Status: Downloaded newer image for portainer/portainer-ce:2.21.5
b2a5584f0cbfc9af9aac7cbc1fd44c5dbfdfe6fd8e...2.21.5:bcca3437b9796b
[root@localhost ~]#
```

Configuration de portainer

Pour se connecter à portainer, il suffit de se connecter à http://<IP_DE_VOTRE_SERVEUR>:9000.

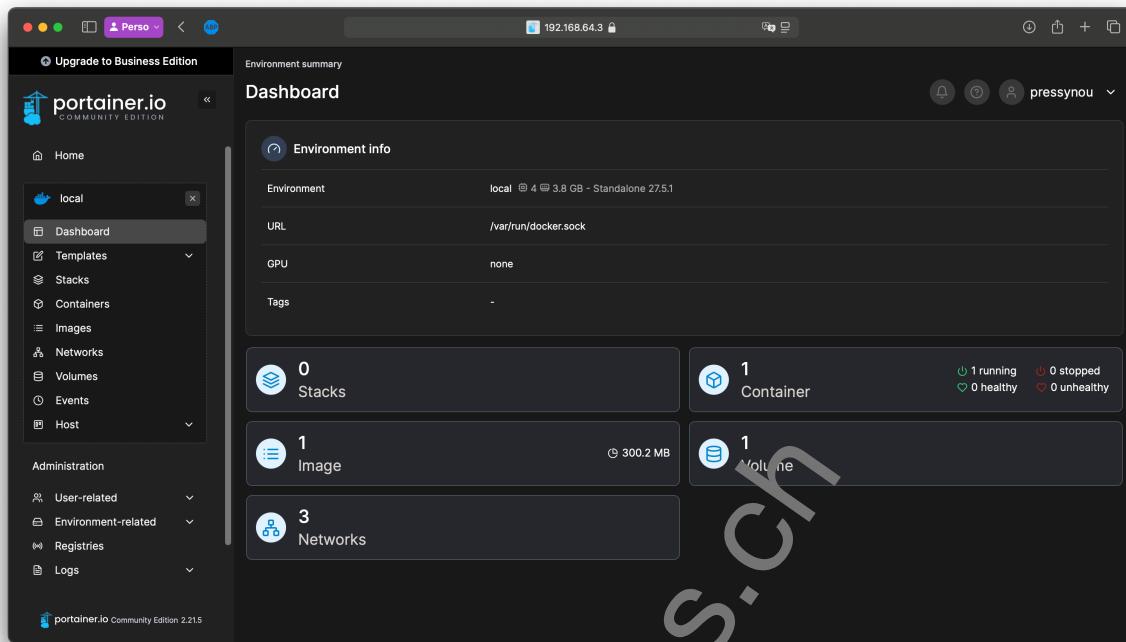


Installation, configuration et utilisation de Docker

The screenshot shows the Portainer Container list interface. On the left, there's a sidebar with navigation links like Home, Dashboard, Templates, Stacks, Containers (which is selected), Images, Networks, Volumes, Events, and Host. The main area displays a table of containers. One row is highlighted for the container named "portainer", which is listed as "running". The table includes columns for Name, State, Stack, Image, Created, IP Address, Published Ports, and Ownership. The ownership is listed as "administrators". There are also buttons for Quick Actions, Restart, Kill, Pause, Resume, and Remove.

The screenshot shows the Portainer Environments interface. The sidebar has links for Home, Environment (None selected), Administration, and various sub-links under User-related, Environment-related, Registries, Logs, Notifications, and Settings. The main area shows the "Environments" section with a table. One row is selected for the "local" environment, which is "Up" and was created on "2025-02-11 08:30:53". It's a "Standalone" type using "/var/run/docker.sock". The table includes columns for Platform, Connectio..., Status, Tags, Groups, Agent Ver..., and Sort By. The "local" environment is shown with 0 stacks, 1 container, 1 image, 4 CPU, 3.8 GB RAM, and is Disconnected. There are buttons for Live connect, Edit, and Delete.

Utilisation de portainer



Installation de Nginx Proxy Manager

Création du fichier compose

```
1. mkdir docker-compose.yaml
```

The screenshot shows a terminal window titled "pressynou — pressynou@localhost:~ — ssh root@192.168.64.3 — 80x24". The command "nano docker-compose.yaml" is running, displaying the following YAML configuration:

```
GNU nano 5.6.1                               docker-compose.yaml
services:
  app:
    image: 'jc21/nginx-proxy-manager:latest'
    restart: unless-stopped
    ports:
      - '80:80'
      - '81:81'
      - '443:443'
    volumes:
      - ./data:/data
      - ./letsencrypt:/etc/letsencrypt
```

The terminal includes a status bar at the bottom with keyboard shortcuts for nano editor commands.

Installation, configuration et utilisation de Docker

Lancement du container

```
pressynou — pressynou@localhost:~ — ssh root@192.168.64.3 — 80x24
[[pressynou@localhost ~]$ sudo docker compose up -d
[+] Running 3/36
  ▲ app [..] Pulling ] 5.6s
```

```
pressynou — pressynou@localhost:~ — ssh root@192.168.64.3 — 80x24
[[pressynou@localhost ~]$ sudo docker compose up -d
[+] Running 36/36
  ✓ app Pulled 92.9s
  [+]
  ✓ Network pressynou_default Created 0.4s
  ✓ Container pressynou_app-1 Started 0.6s
[pressynou@localhost ~]$
```

Installation, configuration et utilisation de Docker

The screenshot shows the Portainer.io interface. On the left, there's a sidebar with navigation links: Home, local, Dashboard, Templates, Stacks, Containers (which is selected), Images, Networks, Volumes, Events, and Host. Below that is an Administration section with User-related, Environment-related, Registries, and Logs. At the bottom of the sidebar is the text "portainer.io Community Edition 2.21.5". The main area is titled "Container list" and shows two containers: "portainer" (running, portainer/portainer-ce:2.21.5) and "pressynou-app-1" (running, jc21/nginx-proxy-manager:latest). There are buttons for Start, Stop, Kill, Restart, Pause, Resume, and Remove. A "Quick Actions" dropdown is also visible. The top bar shows the IP address 192.168.64.3.

Connexion à Nginx Proxy Manager

The screenshot shows the Nginx Proxy Manager landing page. It features a large "Congratulations!" heading, followed by the message "You've successfully started the Nginx Proxy Manager." Below that, it says "If you're seeing this site then you're trying to access a host that isn't set up yet." and "Log in to the Admin panel to get started." At the bottom, it says "Powered by Nginx Proxy Manager". The top bar shows the IP address 192.168.64.3.

Installation, configuration et utilisation de Docker

Pour se connecter au panel d'administration, il suffit de se connecter à `http://<IP_DE_VOTRE_SERVEUR>:81`.
Mots de passe par default : changeme

The image consists of two vertically stacked screenshots of the Nginx Proxy Manager application. Both screenshots are taken from a Mac OS X desktop environment, as indicated by the window title bar and icons.

Top Screenshot (Login Screen):

- Window Title: Portainer | local
- Address Bar: 192.168.64.3
- Content: "Login to your account" form with:
 - Nginx PROXY MANAGER logo and version v2.12.3
 - Email address input field: admin@example.com
 - Password input field (obscured)
 - Sign in button

Bottom Screenshot (Dashboard):

- Window Title: Portainer | local
- Address Bar: 192.168.64.3
- Content:
 - Welcome message: "Hi Admin"
 - Four summary cards:
 - Proxy Hosts: 0
 - Redirection Hosts: 0
 - Streams: 0
 - 404 Hosts: 0
 - Navigation menu: Dashboard, Hosts, Access Lists, SSL Certificates, Users, Audit Log, Settings
 - User info: "Admin" (Administrator)
 - Footer: v2.12.3 © 2024 jc21.com. Theme by Tabler, Fork me on Github