



Portainer 是Docker的轻量级，跨平台和开源管理UI。Portainer提供了Docker的详细概述，并允许您通过基于Web的简单仪表板管理容器，图像，网络 and 卷。它最初是Docker UI的分支。但是，开发人员现在已经重写了几乎所有的Docker UI原始代码。他还彻底修改了UX，并在最新版本中添加了更多功能。截至目前，它已经引起了用户的极大关注，并且现在已经有超过一百万的下载量！它将支持GNU / Linux，Microsoft Windows和Mac OS X。

1. 安装Portainer

Portainer的安装非常简单，将花费不到一分钟的时间。Portainer完全支持Docker 1.10及更高版本。要安装Docker，请参考以下链接。

- [如何在CentOS上安装](#)
- [如何在Ubuntu上安装](#)
- [查看镜像](#)

安装Docker之后，运行以下命令以获取最新的Portainer映像。

```
[root@localhost ~]# docker search portainer
```

```
[root@localhost ~]# docker search portainer
NAME                                DESCRIPTION                                STARS                                OFFICIAL                                AUTOMATED
portainer/portainer                Making Docker management easy. https://porta... 1395
portainer/agent                    An agent used to manage all the resources in... 48
portainer/templates                App Templates for Portainer http://portainer... 16
lihaixin/portainer                 docker ui                                8 [OK]
portainerci/portainer              Portainer images automatically created via P... 4
portainer/golang-builder            Utility to build Golang binaries.            4 [OK]
greenled/portainer-stack-utils     Bash scripts to deploy/undeploy stacks in a ... 3 [OK]
stefanscherrer/portainer           Portainer for Windows 1809 and some older In... 2
dockerframework/portainer          Portainer (Simple Management UI for Docker) ... 1 [OK]
j0nnylester/portainer              portainer running on a Raspberry Pi         1
hassioaddons/portainer             1
softonic/portainer-endpoint         Allows auto register all the swarm nodes in ... 1 [OK]
portainer/base                     Multi-stage build image to create the Portai... 1 [OK]
rancher/portainer-agent            0
hassioaddons/portainer-amd64        0
rodjers/portainer                  Swarm portainer                            0
11384eb/portainer                  0
hassioaddons/portainer-armv7        0
nenadilic84/portainer              0
helloyds/portainer                 0
portainer/angular-builder           Builder image for Portainer frontend.         0 [OK]
ssbkang/portainer                  0
profidata/portainer                Fork of portainer/portainer                  0
portainerci/agent                  Portainer agent images automatically created... 0
webdevsvc/portainer                portainer                                    0 [OK]
[root@localhost ~]# docker pull portainer/portainer
Using default tag: latest
```

- 拉取镜像

```
[root@localhost ~]# docker pull portainer/portainer
Using default tag: latest
latest: Pulling from portainer/portainer
d1e017099d17: Pull complete
292b789be2e4: Pull complete
Digest: sha256:63eb47d4b408c3f39e942368bcbf7e157a2b6e5dda94ffd403a14199e1137133
Status: Downloaded newer image for portainer/portainer:latest
docker.io/portainer/portainer:latest
```

- 查看镜像

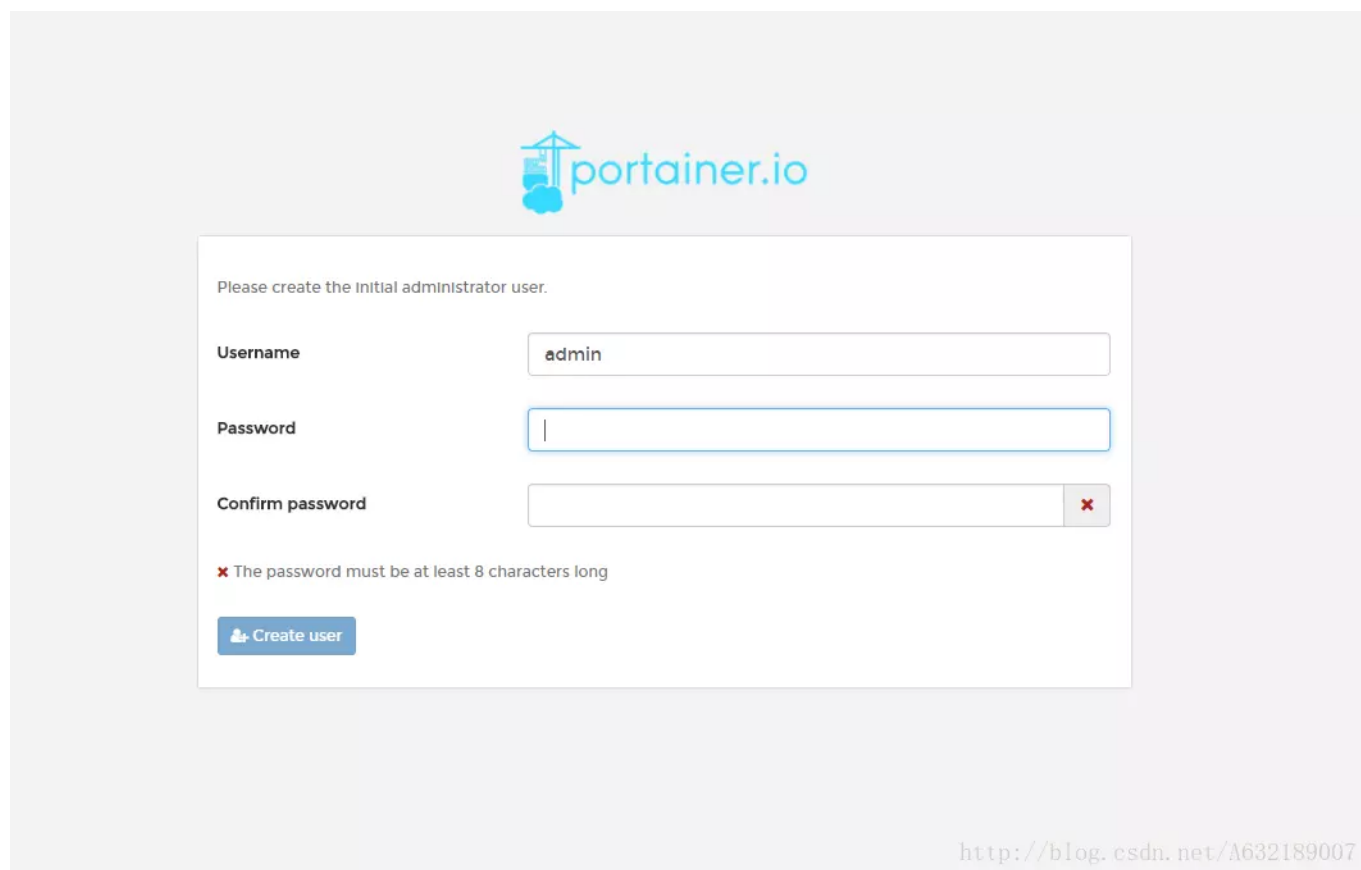
```
[root@localhost ~]# docker images
REPOSITORY                                TAG                                IMAGE ID
CREATED                                SIZE
portainer/portainer                    latest                            d1219c88aa21
2 weeks ago                            80.8MB
```

从上面的输出中可以看到，Portainer的镜像约80 MB，这样我就不会占用更多的RAM和Hdd空间。

2. 启动

```
[root@localhost ~]# docker run -d -p 9000:9000 --restart=always -v
/var/run/docker.sock:/var/run/docker.sock --name prtainer-test portainer/portainer
```

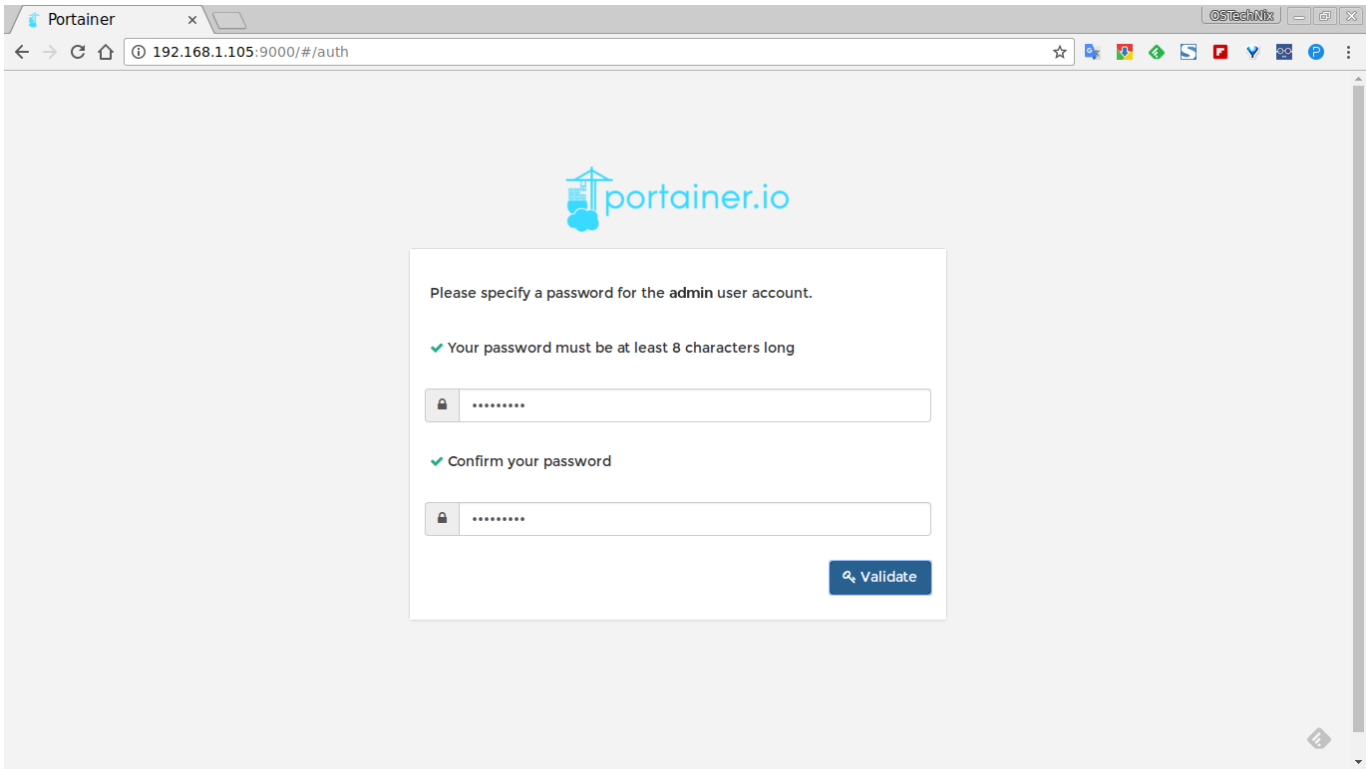
该语句用宿主机9000端口关联容器中的9000端口，并给容器起名为portainer-test。执行完该命令之后，使用该机器IP:PORT即可访问Portainer。



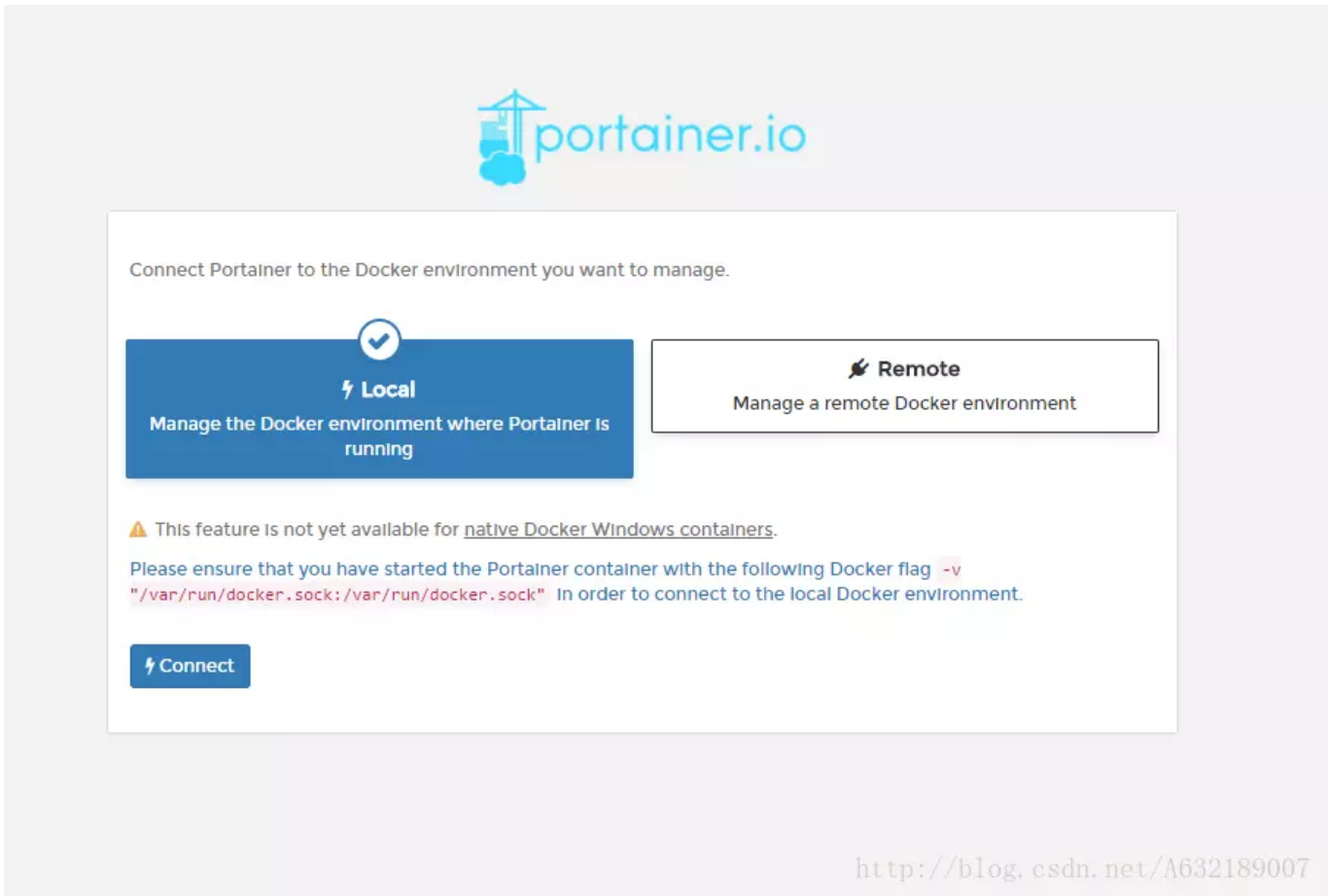
3. 访问

http://IP:9000

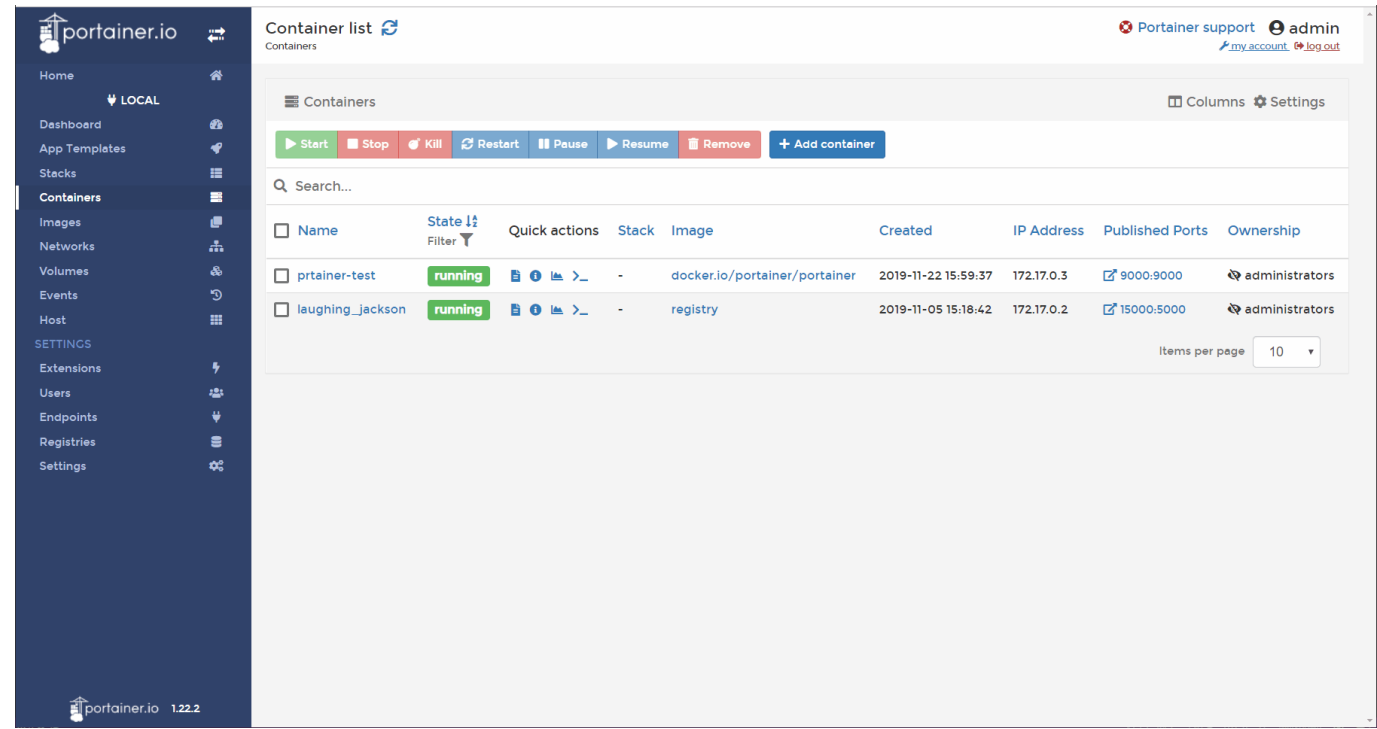
首次登陆需要注册用户，给管理员用户admin设置密码。



单机版这里选择local即可，选择完毕，点击Connect即可连接到本地docker



注意：该页面上有提示需要挂载本地 /var/run/docker.sock与容器内的/var/run/docker.sock连接。因此，在启动时必须指定该挂载文件。

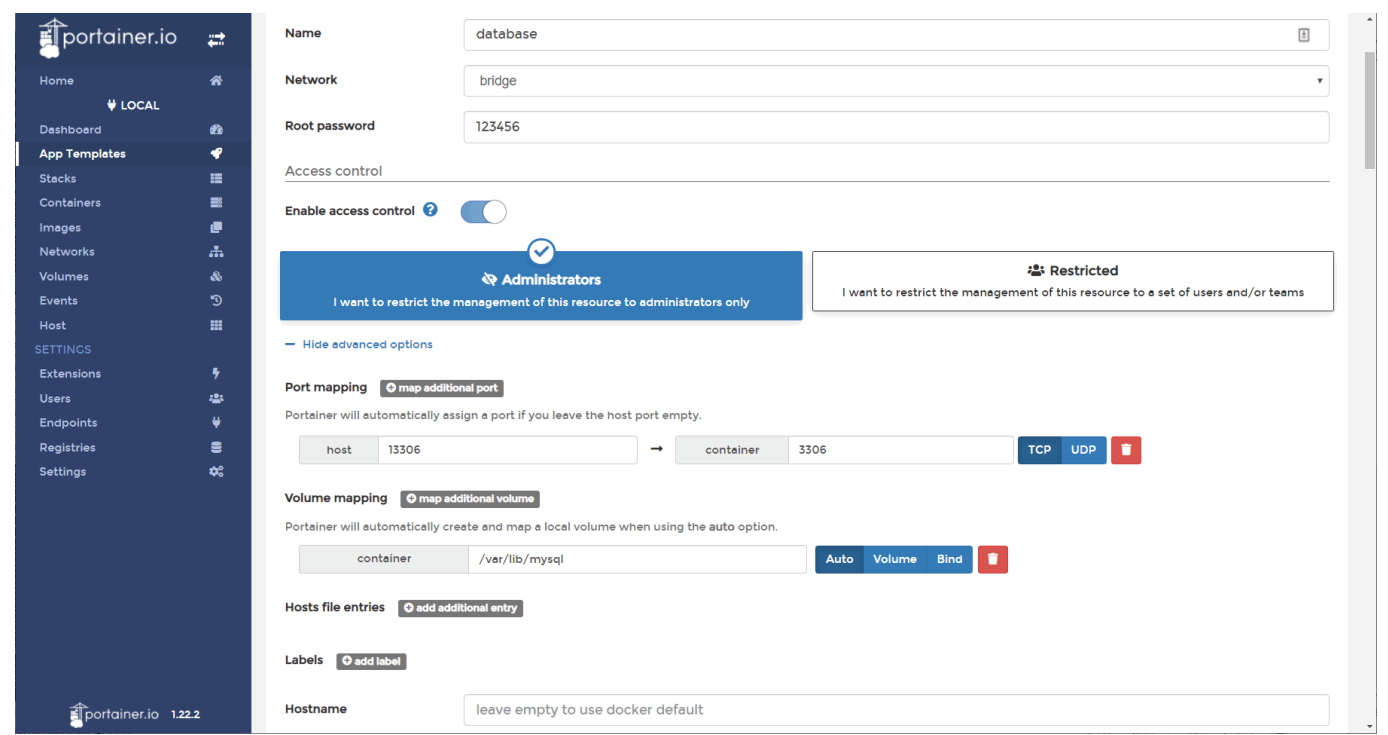


4. 创建容器

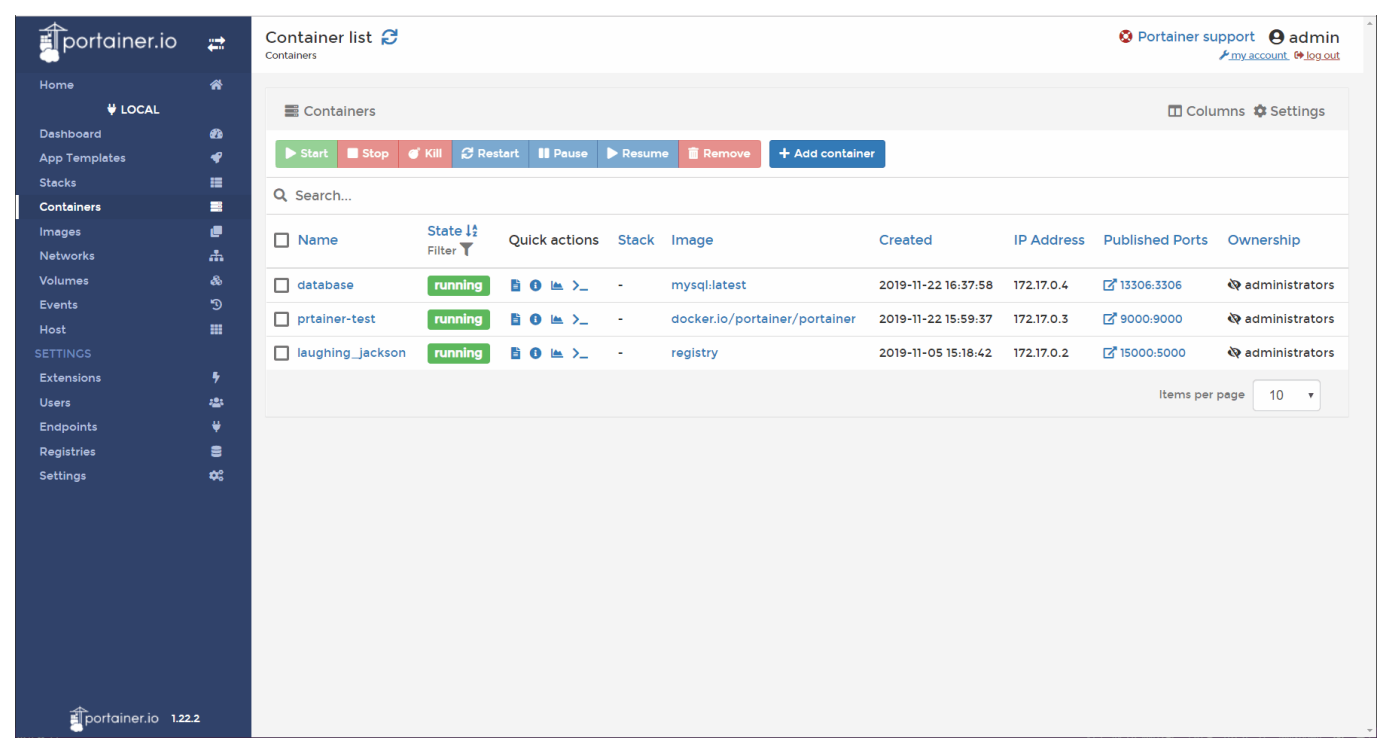
单击左侧的“App Templates”按钮。这将显示一些现成的可用模板，例如Docker映像注册表，Nginx，Httpd，MySQL，WordPress等。

6 / 12

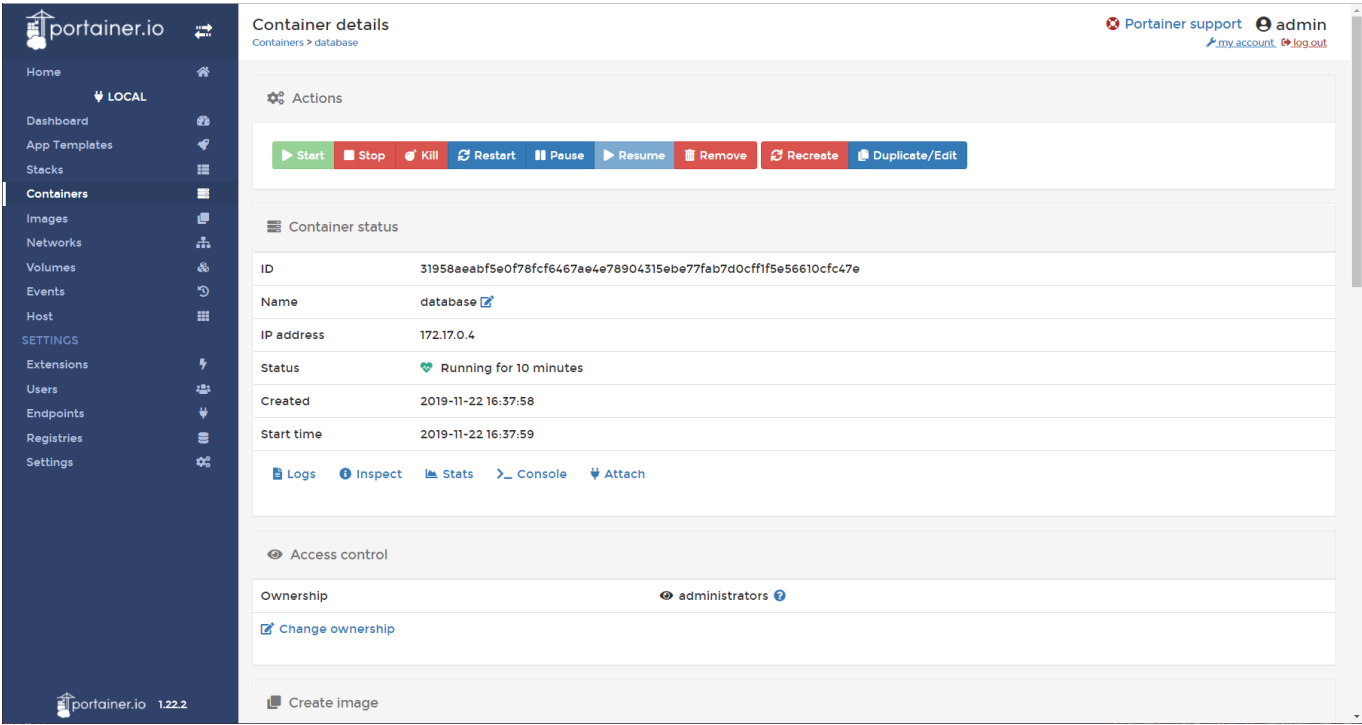
这里让我们启动MySQL容器。为此，请单击MySQL模板。输入容器名称，选择网络类型（例如新娘模式）。单击“Show advanced options”并设置端口号。



点击“Deploy the container”，过程会有点长，耐心等待。

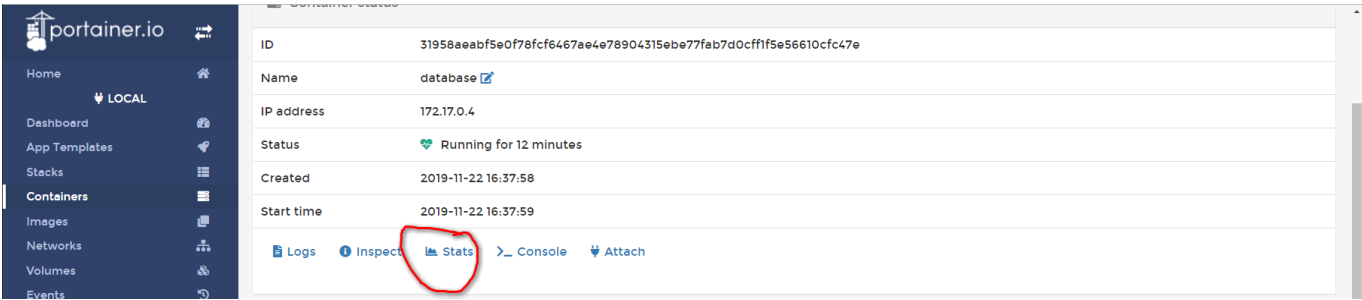


选择容器，点击进入。

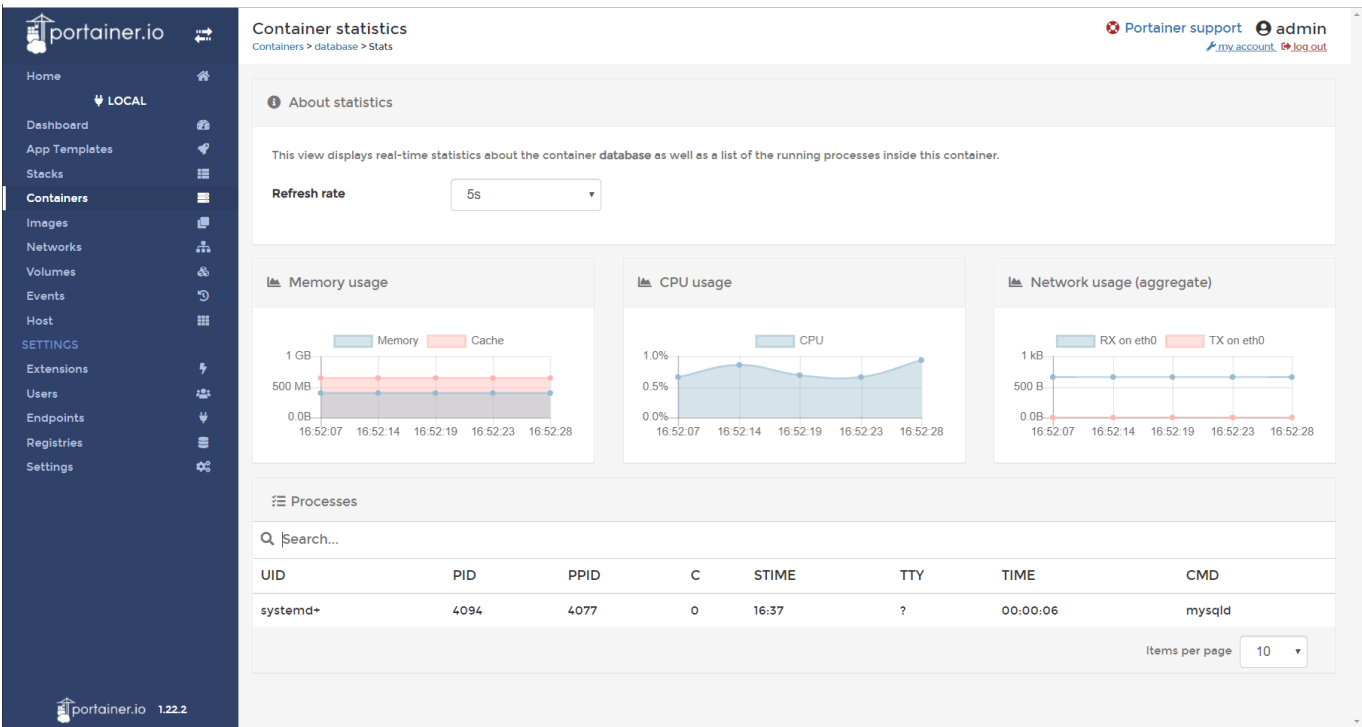


可以这里重新启动，停止，暂停和删除此部分中的容器。

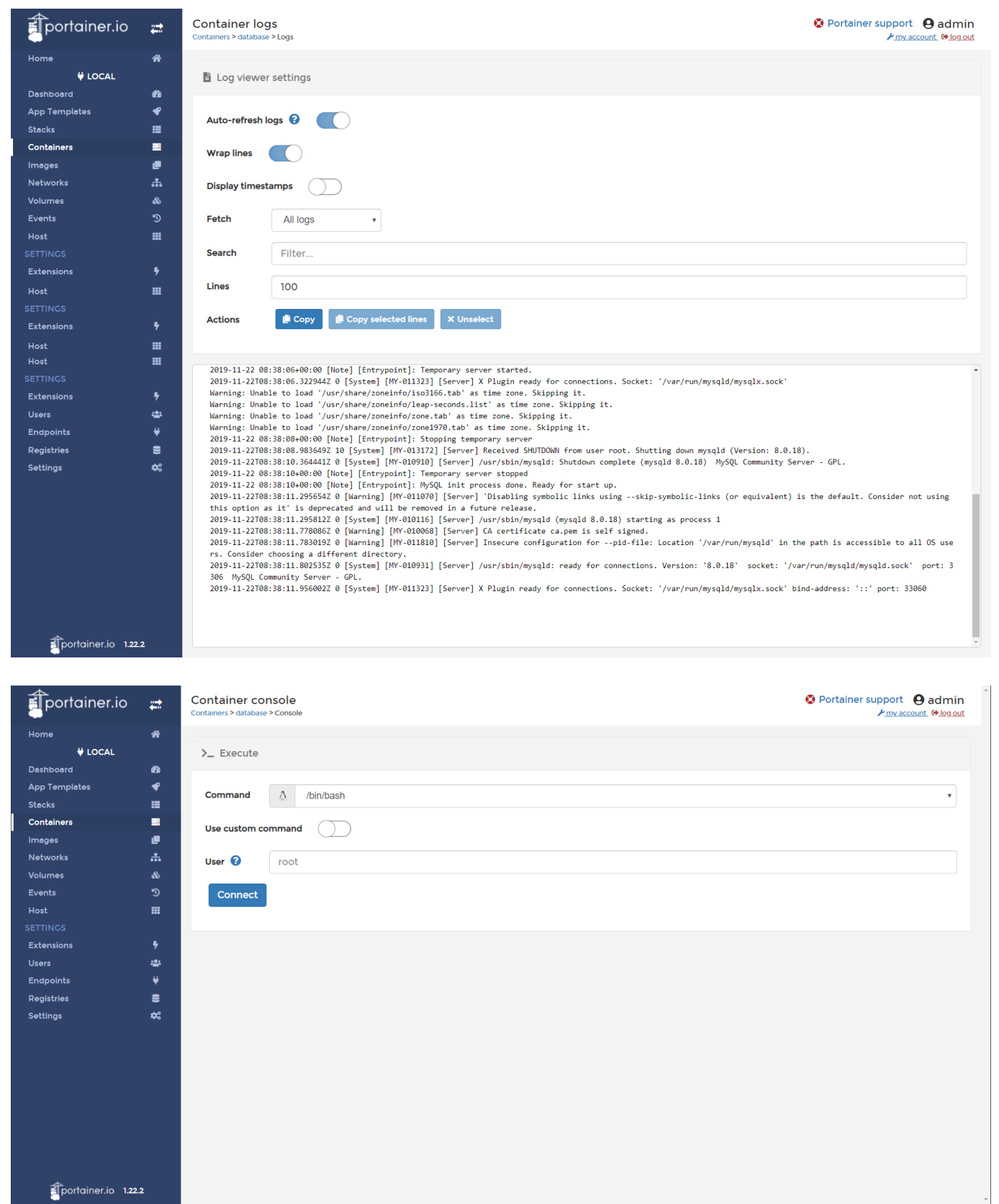
点击“Stats”统计信息按钮以查看新启动的容器中发生的情况。



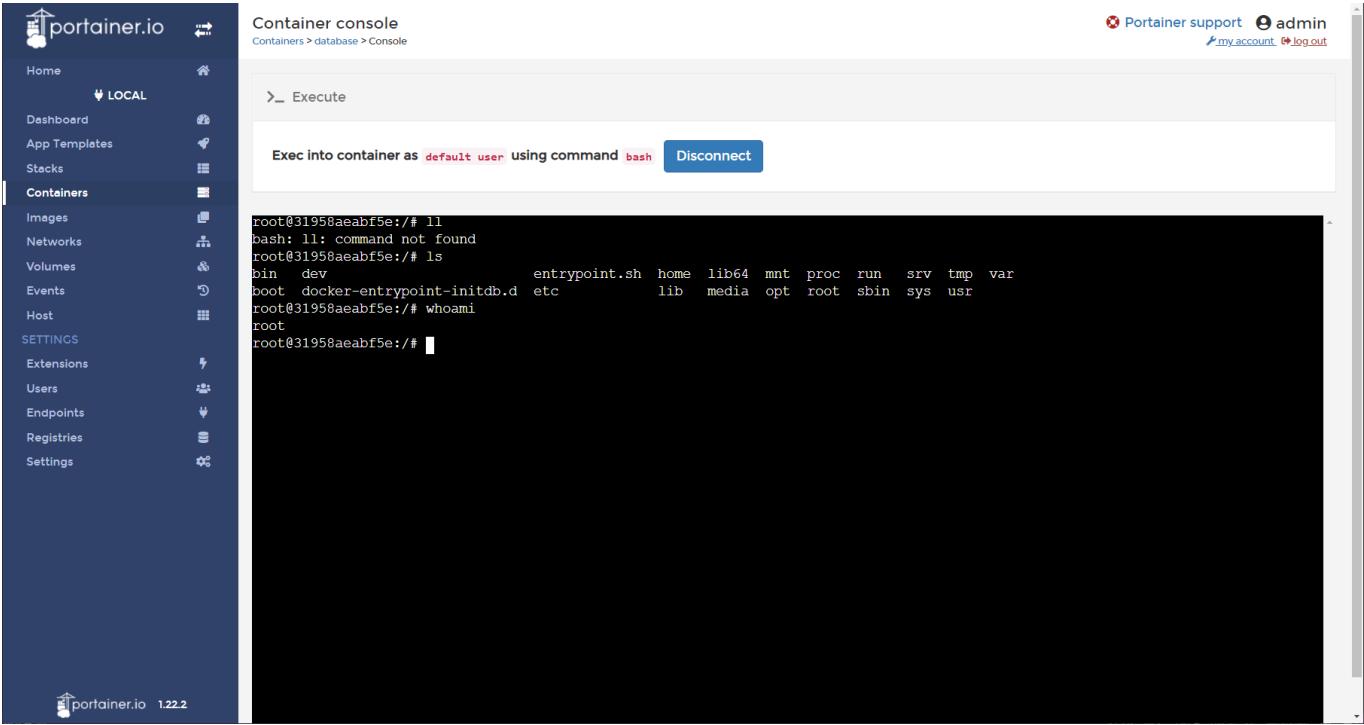
这是我的数据库容器的统计信息



同样，单击“Logs”按钮以查看容器的完整日志详细信息

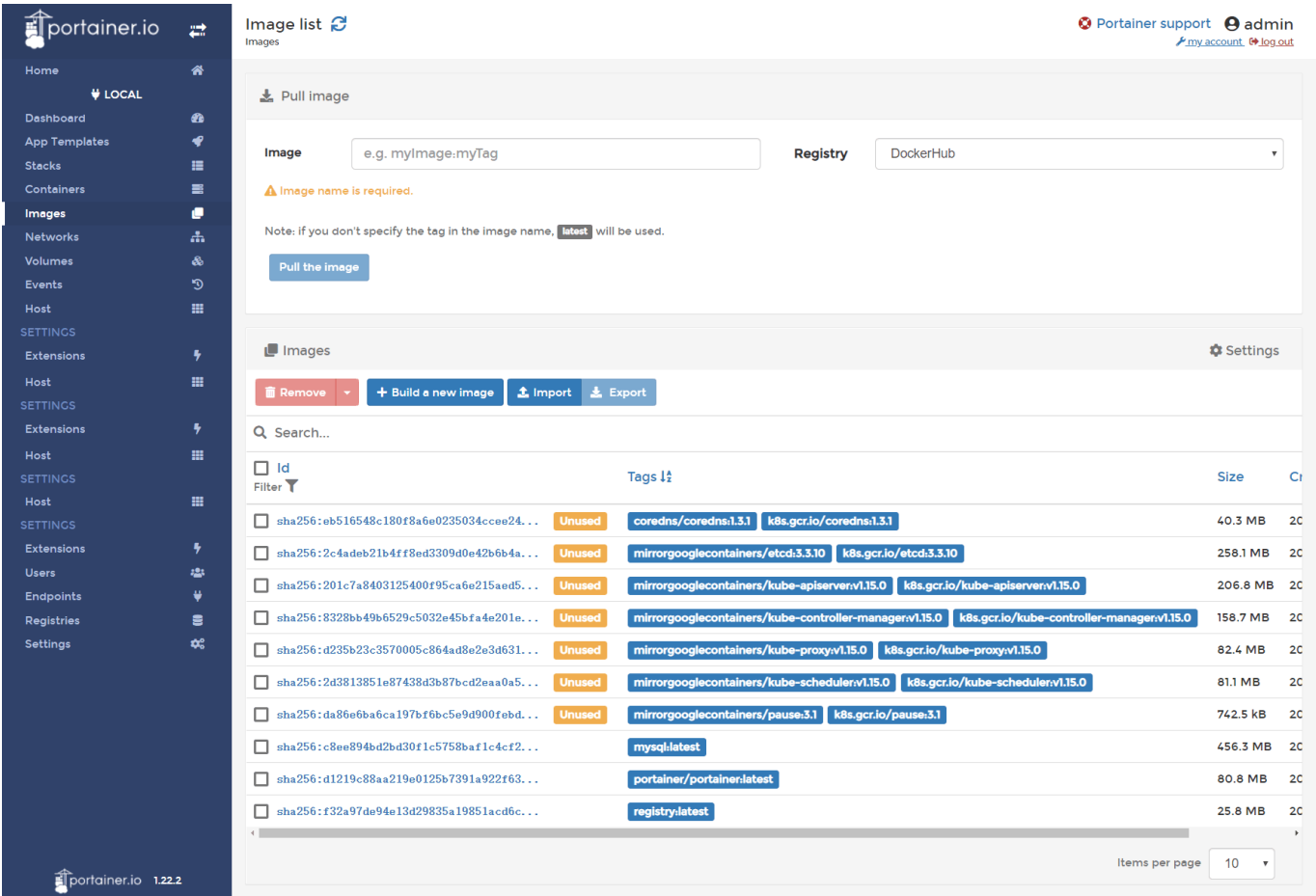


选择命令行管理程序（BASH或SH），然后单击“Connect”按钮



5. Docker镜像

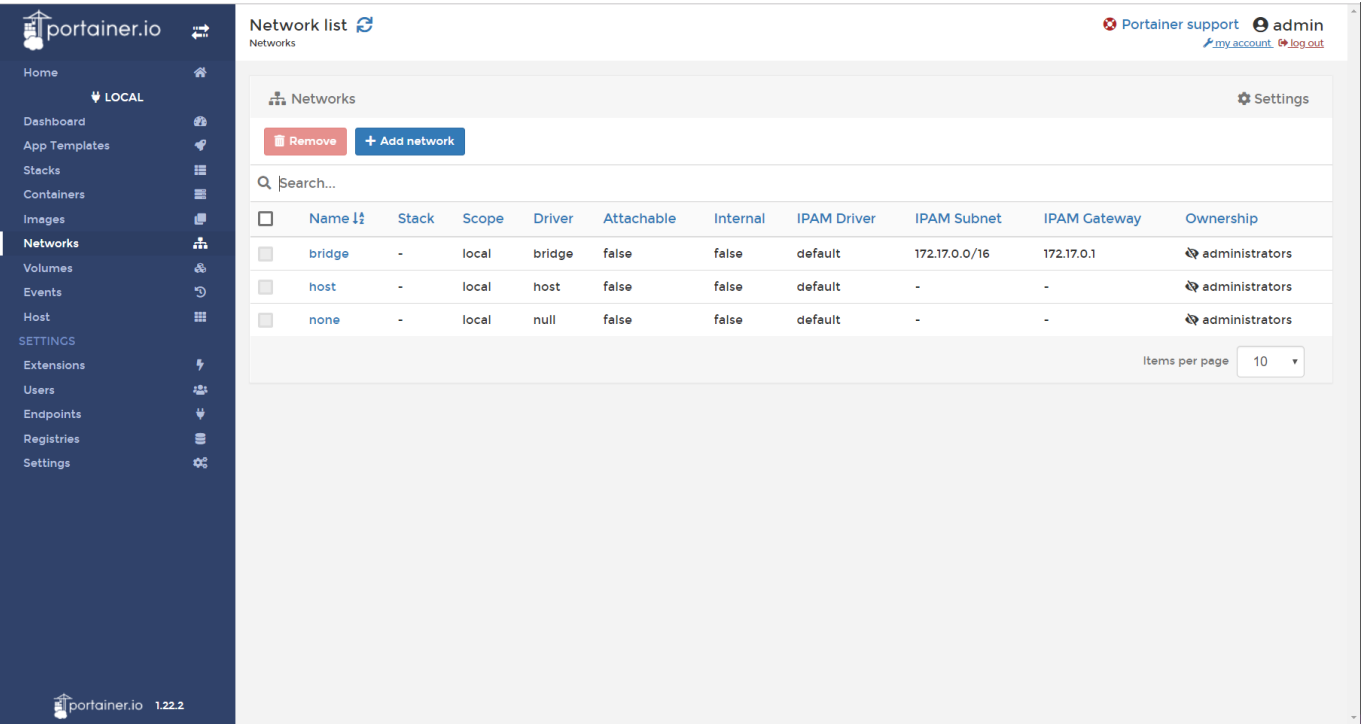
您可以查看已下载的Docker映像的列表。



要删除任何图像，只需选择它，然后单击“Remove”，所选图像将消失。

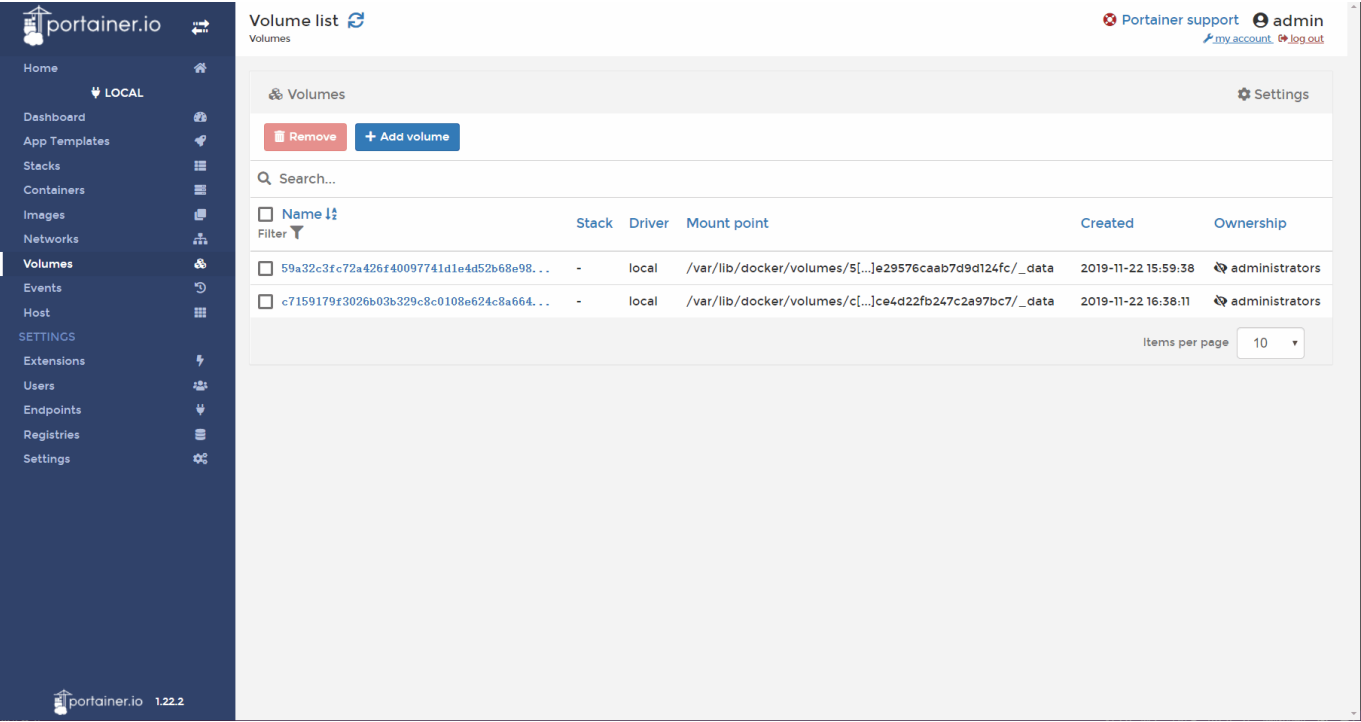
6. 网络（Networks）

网络部分允许您添加新网络，更改网络类型，分配/更改IP地址，删除现有网络。



7. 卷簇（Volumes）

在这里您可以查看现有的docker卷，创建新的docker卷，如果不再需要它们则将其删除。



8. 事件

在这里，我们可以查看到目前为止我们正在做的事情，例如创建新实例，网络，卷等。

portainer.io

Home

LOCAL

Dashboard

App Templates

Stacks

Containers

Images

Networks

Volumes

Events

Host

SETTINGS

Extensions

Users

Endpoints

Registries

Settings

portainer.io 1.22.2

Event list

Events

Portainer support admin

my account log out

Events

Search...

Date	Category	Details
2019-11-22 16:58:24	container	Exec instance started
2019-11-22 16:58:24	container	Exec instance created
2019-11-22 16:53:07	container	Showed running processes for container database
2019-11-22 16:53:03	container	Showed running processes for container database
2019-11-22 16:52:58	container	Showed running processes for container database
2019-11-22 16:52:52	container	Showed running processes for container database
2019-11-22 16:52:47	container	Showed running processes for container database
2019-11-22 16:52:42	container	Showed running processes for container database
2019-11-22 16:52:37	container	Showed running processes for container database
2019-11-22 16:52:32	container	Showed running processes for container database

Items per page 10

< 1 2 3 4 >